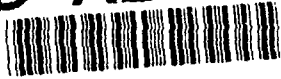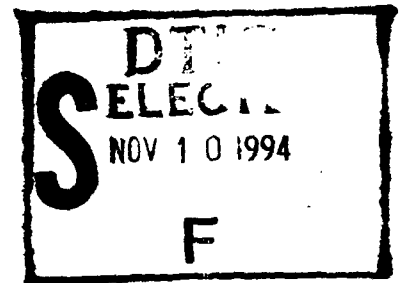AD-A286 082

Software Capability Evaluation (SCE)

Version 2.0
Team Members' Guide

Members of the
CMM-Based Appraisal Project

October 1994

94-34817

Software Capability Evaluation (SCE)

Version 2.0
Team Members' Guide

Members of the
CMM-Based Appraisal Project

A-1
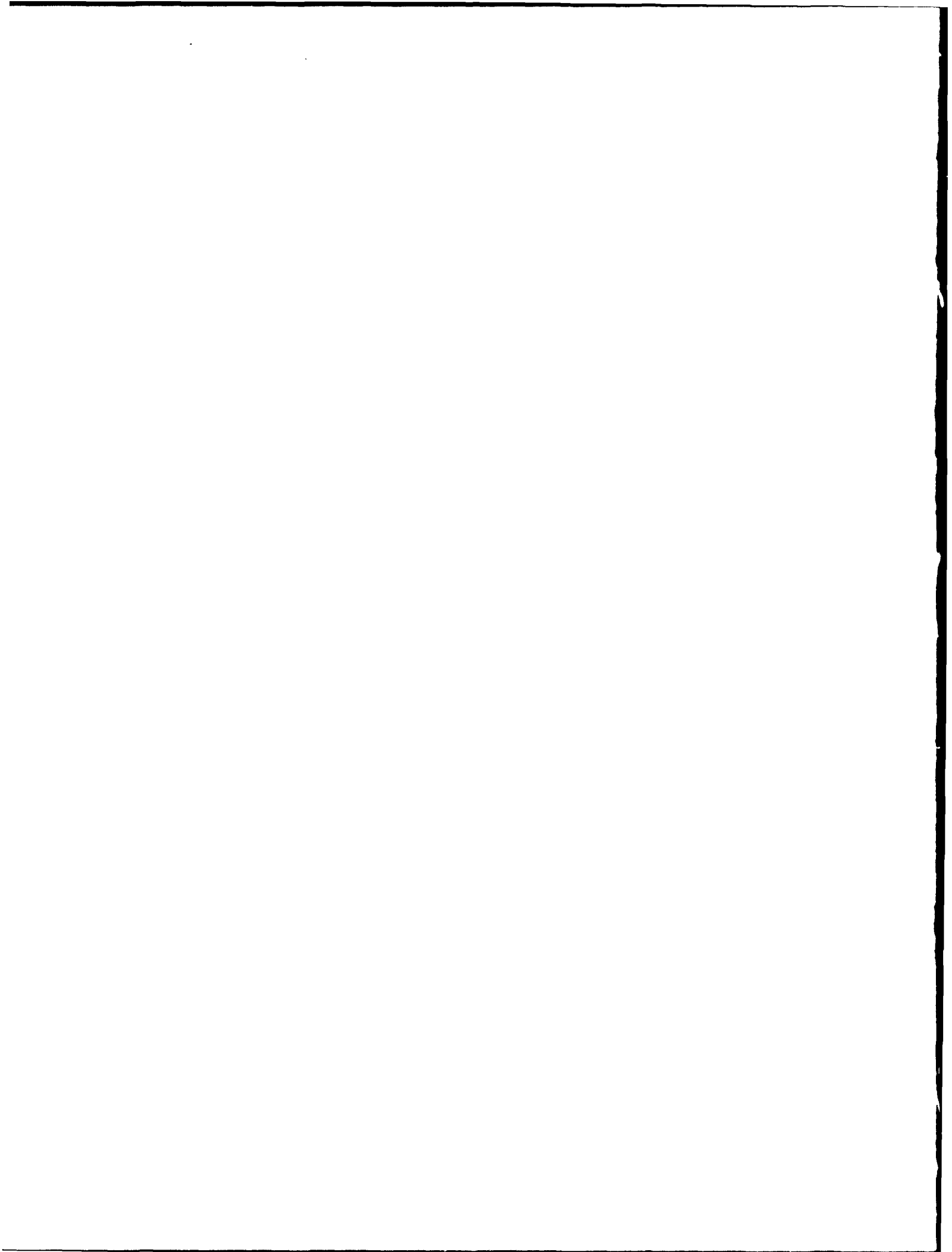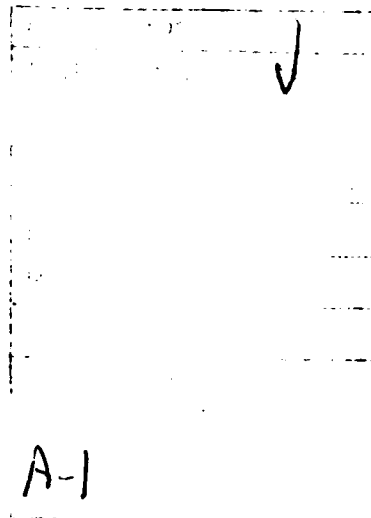
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

This report was prep   for the
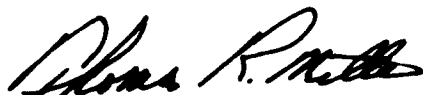
SEI Joint Program Office
HQ ESC/ENS
5 Eglin Street
Hanscom AFB, MA 01731-2116

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

Review and Approval

This report has been reviewed and is appr. . . . . r publication.

FOR THE COMMANDER

Thomas R. Miller, Lt Col, USAF
SEI Joint Program Office

This work is sponsored by the U.S. Department of Defense.

Table of Contents

Part 2   Instructions and Guidelines

Part 3   Appendices

Appendix A   Steps of the SCE Method

Appendix B   Maturity Model

Appendix C   Attribute Definitions

Appendix D   Sample Forms

List of Guidelines

List of Instructions

List of Figures

List of Tables

Part 1   Introduction

Chapter 1   About This Guide

Chapter 1   About This Guide

Software Capability Evaluation (SCE) is a method for independently evaluating the software process of an organization to gain insight into its software development capability. The method is defined in the report *Software Capability Evaluation Version 2.0 Method Description*. [SCE 93b]

This document, the *Software Capability Evaluation Version 2.0 Team Member's Guide*, is intended for use by members of teams that will be conducting an SCE. The guide provides detailed step-by-step instructions and heuristic information to assist an SCE team in preparing for and conducting an evaluation.

Purpose

The *Team Member's Guide* is provided as a supplement to the *SCE Team Training* offered by the Software Engineering Institute. It is intended to be used in conjunction with the *Method Description* and with the *Software Capability Evaluation Version 2.0 Implementation Guide* [SCE 93a]. The *Team Member's Guide* provides information to reinforce the training and it will serve as a reference for use after the training.

The guide provides convenient access to information needed while preparing for and conducting a site visit. In addition to the detailed instructions it includes guidelines and heuristic information to help the team in gathering and evaluating information. It also includes checklists to help in planning for and conducting site visits. The guide is intended to be the primary reference on the SCE method that teams will need to carry with them to site visits.

Scope

The *Team Member's Guide* provides detailed step-by-step instructions for all of the activities that are normally performed by an SCE team.

The *Method Description* [SCE 93b] defines five phases in the SCE method: 1) Evaluation Start, 2) General Preparation, 3) Specific Preparation, 4) Site Data Collection, and 5) Findings. Normally, the Evaluation Start phase is performed by the sponsoring agency and the other four phases are performed by the SCE team.

The *Team Member's Guide* concentrates on the activities in phases 2 through 5. Detailed instructions are provided for each of the steps within these phases. Information is provided to help teams plan and conduct the activities associated with these steps.

Some information is provided about the Evaluation Start phase and about other activities that the sponsoring agency performs prior to the start of the SCE (➡Chapter 4). This information is included only to provide a context for the team's activities. It does not provide the level of detail needed to understand how to perform the activities. Refer to the *Implementation Guide* [SCE 93a] for more detailed information.

This version of the guide is compatible with the *Software Capability Evaluation Version 2.0 Method Description* [SCE 93a] and with the *Capability Maturity Model for Software, Version 1.1* [Paulk 93a].

Intended Audience

The *Team Member's Guide* is intended for use by members of teams that will be conducting an SCE. It assumes that the team members have also attended the *SCE Team Training*. This guide alone does not adequately prepare a team to conduct an SCE.

The guide assumes that the team collectively has experience in software system development and acquisition. The effectiveness of the SCE method depends

in a large part on the experience of the team members who must be able to make judgements about the information they collect. The recommended experience for team membership is listed in the *Implementation Guide*. [SCE 93a]

Section 1   How the Guide Is Organized

This guide is divided into three parts.

Part 1 provides background information that the team members should understand before starting an SCE.

- **Chapter 2, Method Overview**, provides a summary of the SCE method and an explanation of the underlying principles. This section will help the team members understand the objectives of each step and see how the steps relate to each other.

- **Chapter 3, Team Development**, provides information on how to build and train a team that will be able to effectively conduct an SCE.

Part 2 provides explicit instructions along with guidelines and heuristic information to aid the team in each step of planning for and conducting the site visit.

- **Chapter 4, Before the SCE Team Begins**, describes the preparations that should be completed by the time the team is assembled. These activities are normally performed by the sponsoring agency and are not considered to be part of the team's responsibility. They are described briefly to provide the context for understanding the team's activities.

- **Chapter 5, Preparing for the Site Visit**, provides detailed instructions for the preparation the team must do to get ready for the site visit.

- **Chapter 6, Conducting the Site Visit**, provides detailed instructions for arranging and conducting a site visit.

- **Chapter 7, Interviews**, provides general information and guidance on how to plan for and conduct interviews. Information in this section is generally applicable across several of the steps described in Chapter 5 and Chapter 6.

- **Chapter 8, Document Review**, provides general information and guidance on conducting the document reviews. Information in this section is generally applicable across several of the steps described in Chapter 6.

- **Chapter 9, Team Caucus**, provides general information on the team caucus process and on how to reach a consensus on the information collected.

The information in Part 2 is of three types: instructions for performing the formal phases and steps of the SCE method, instructions for the planning and supporting activities, and descriptions of the special techniques (interviewing, document reviewing, and holding a team caucus) which apply to more than one step of the method.

The information is arranged in the order in which the team is likely to perform the activities. Sections containing instructions for planning an supporting activities are distributed among the sections describing the formal phases and steps. Sections describing the special techniques are grouped at the end. Figure 1-1 shows how the information is organized relative to the activities the team performs.

Part 3 includes all of the appendices. The appendices contain material that the team members will need to refer to frequently.

- **Appendix A, Steps of the SCE Method,** contains a table showing the name and purpose of each step in the SCE method and flow diagrams for each phase of the method to show how the steps are related.

- **Appendix B, Maturity Model,** contains material extracted from the CMM which the team will need to refer to most often.

- **Appendix C, Attribute Definitions**, contains the definitions of the attributes used to describe products and projects on the various profiles and worksheets used in the SCE preparation activities.

SCE Activities

Part 2
Instructions and Guidelines

Implement SCE in the Acquisition Process

Request Information From the Development Organizations

Phase 1

Evaluation Start

Phase 2

General Preparation

Phase 3

Specific Preparation

Arrange for the Site Visit

Phase 4

Site Data Collection

Phase 5

Findings

Time

Chapter 4
Before the Team Begins

Section 1

Section 2

Section 3

Chapter 5
Preparing for the Site Visit

Section 1

Section 2

Chapter 6
Conducting the Site Visit

Section 1

Section 2

Section 3

Chapter 7
Interviews

Chapter 8
Document Reviews

Chapter 9
Team Caucus

Phase of the formal SCE method

Planning or supporting activity

SCE team activities

Figure 1-1: Organization of the Information in Part 2

- **Appendix D, Sample Forms**, contains a description of the forms used throughout the SCE process with samples of completed forms.

- **Appendix E, Blank Forms**, provides blank copies of the forms used throughout the SCE process which may be used as reproduction masters.

- **Appendix F, Subprocess Area Selection Tables**, contains information which may be used to help select critical subprocess areas to investigate.

- **Appendix G, Look-For Tables**, provides information to guide the teams in collecting information during the site visit.

- **Appendix H, Checklists**, contains checklists to assist teams with planning and performing the various activities of an SCE.

- **Appendix I, Sample Entry Briefing**, contains an example of an entry briefing which a team might present to the development organization at the start of the site visit.

- **Appendix J, Sample Exit Briefing**, contains an example of an exit briefing which a team might present to the development organization at the conclusion of the site visit.

- **Appendix K, Glossary**, contains definitions of important terms used in this document.

- **Appendix L, Bibliography**, contains a list of the other documents referred to in this document.

Section 2   How to Use this Guide

New SCE Teams

For new SCE teams, the guide provides detailed instructions for each of the activities a team normally performs to prepare for and conduct an SCE. It will be a reference to help clarify information that was not understood in the training or that has been forgotten.

The guide includes information that will help a new team take advantage of lessons learned from other teams. It also includes useful heuristic information that will help a team plan an SCE for the first time. After conducting a few SCEs, the team may develop their own heuristics to use instead.

Experienced SCE Teams

Experienced team members could benefit from reading through the guide even though they feel they understand the method already. The SCE method has evolved as experience has shown areas that needed to be improved. If it has been some time since the team was trained, they may not be aware of the changes to the method. Also the guidelines and heuristic information may bring out points that the team has not considered or discuss problems that the team may not have encountered yet. Primarily though, the experienced team should find that the appendices contain the information that they need to refer to most frequently when conducting an SCE.

Before Starting an SCE

Before starting an SCE, the team should become familiar with the material in Chapter 2 and Chapter 3. The team should use the material in Chapter 4 to review the preparations that should have been completed before the team begins its activities.

Planning and
Preparation

During the planning and preparation phase the team will
need to refer primarily to the material in Chapter 5 and in
the appendices.

Site Visit

Before the site visit, the team members should become
familiar with the material in Chapter 6, Chapter 7,
Chapter 8, and Chapter 9. During the site visit the team
the team will primarily use the appendices with some
occasional references to guidelines in other sections.

Section 3   Conventions Used

Each part of the document is organized into chapters and sections within chapters. Sections which describe one of the formal phases of the SCE method are further broken down by the steps within that phase. Sections and steps may contain major subsections, which are identified by a heading at the top of a page, and minor subsections, which are identified by a heading in the left margin.

Running Headings

The running heading at the top of each page shows where you are in the document. The top two line are always the current chapter and section. If the section is a description of one of the formal phases of the method, the third line is the current step number. If the page is part of major subsection, then the bottom line contains the subsection heading. Figure 1-2 shows an example of a running heading containing chapter, section, step, and subsection headings.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 4   Create Experience Table
Guidelines   How to Determine Experience Mismatches

Guidelines   **How to Determine Experience Mismatches**

The SCE team evaluates the relevant experie
development organization by comparing the I
Profiles and the Proposed Project Profile that

Figure 1-2: Sample Running Heading

Descriptions of the
Formal Phases of the
Method

Each of the sections which describe a formal phase of the method have a standard format. The following information about the phase is provided:

- The objective of the phase.

- The participants who normally perform the activities of the phase.

- The time normally required to complete the activities of the phase.

- The entry criteria - the conditions which must be met before starting the activities of the phase.

- The activities of the phase - the formal steps within the phase.

Both the entry criteria and the activities are in the format of checklists. The entry criteria checklist may be used to record that the preceding activities have been completed. The activities checklist may be used to record that the steps of the phase have been completed. Both lists contain page numbers which refer to where the activities are described.

Descriptions of the
Formal Steps of the
Method

Following the description of the phase, instructions are provided for each of the steps within the phase. The descriptions of the steps have the following format:

- The objective of the step

- A brief description of the step

The description of the step may then be followed by major subsections which contain detailed instructions and guidelines for performing the activities of the step.

Index to Related Topics   In some cases, the information needed to perform a step may be found in other chapters. An index to the applicable information is provided in the format shown below.

Chapter 6	Guidelines, What to Look For During Initial Document Review	page 6-16
Chapter 8	Section 1, Types of Documents	page 8-2
	Section 2, What to Review	page 8-4
	Section 3, What to Look For	page 8-6
	Guidelines, Reviewing Organization Level Policies and Directives	page 8-8
	Guidelines, Reviewing Project Standards and Procedures	page 8-10
	Guidelines, Reviewing Evidence of Process Activity	page 8-12
	Section 4, How to Record Results	page 8-14

Checklist   A checklist may have two levels of information as shown below. If the first level checklist item has one or more subordinate checklist items, then the first level item is complete when all of the subordinate items have been completed.

❑ This is a first level checklist item.
 	
   ❑ This is a subordinate checklist item.
 	
   ❑
	page reference

Guidelines

Guideline information is organized as a list of rules or significant points that the team should consider. The key idea is stated briefly in one or two sentences. The key idea is preceded by the symbol ☞ to make it easy to identify on the page.

Each guideline may be followed by additional text or a subordinate list of minor points or examples which elaborate on the key idea. The following shows the typical format for a list of guidelines:

☞ This is a statement of a rule or a significant point that the team should consider.

This is a paragraph which elaborates on the key idea.

☞ This is another rule

  • This is a list of minor points or examples.

  •

  •

Embedded Page References

The symbol (➡page ##) is used to imbed cross-references within the text. They may be ignored when reading the document. They point to the chapter and page where more information may be found.

Chapter 2   Method Overview

Chapter 2   Method Overview

This chapter provides a review of the key concepts that an SCE team member should understand in order to use the method effectively. It describes some of the underlying principles on which the method is based. The section is intended to provide a framework that will help someone new to SCE to understand the objectives of each activity and to put the various activities in perspective.

What Is SCE?

SCE is a method for independently evaluating the software process of an organization to gain insight into its software development capability. The SCE method is defined in the report *Software Capability Evaluatio; Version 2.0 Method Description* [SCE 93b].

How IS SCE Used?

SCE has two primary applications:

- **Source Selection**: SCE is used to evaluate the software process maturity of the potential offerors to identify the offerors that present the least risk.

- **Contract Monitoring**: SCE is used to evaluate the software process maturity of the contractor after selection to identify areas of risk or periodically over the course of the contract to encourage process improvement.

SCE has another purpose that goes beyond the specific acquisition for which it is being used. As more agencies use SCE as part of the source selection or contract monitoring process, development organizations will be motivated to improve their software development processes in order to remain competitive.

Section 1   Fundamental Concepts

An organization's *software process* is the "set of activities, methods, practices, and transformations that are used to develop and maintain software and its associated products (e.g., project plans, design document, code, test cases, and user manuals)" [Paulk 93b].

Software processes vary widely from organization to organization. Knowledge of software process and process management abilities range from almost non existent to being well understood, clearly defined, instrumented, and controlled. The state of an organization's software process and process management capabilities may represent a significant risk to its ability to perform well on software contracts. In recognition of this fact, many development organizations and procurement agencies have been developing techniques to define, measure, and improve software process.

Capability Maturity Model

The SCE method uses the *Capability Maturity Model for Software, Version 1.1* (CMM v1.1) as a basis for investigating and evaluating an organization's software process. CMM v1.1 presents a framework for expressing the process maturity of an organization. It describes five levels of process maturity and characterizes the behavior most often observed in organizations operating at each maturity level. CMM v1.1 is contained in the following documents:

- *Capability Maturity Model for Software, Version 1.1* [Paulk 93a] contains an introduction to the model, descriptions of the five maturity levels, an operational definition of the CMM and its structure, a discussion of how organizations can use the maturity model, and some remarks on the future directions of the CMM.

- *Key Practices of the Capability Maturity Model, Version 1.1* [Paulk 93b] contains the key practices that correspond to the Key Process Areas at each maturity level of CMM v1.1 and information to help interpret the key practices.

The focus of CMM v1.1 is on software *process*. It does not address all of the issues that are important to successful projects.

For example, it does not evaluate the skills of the organization's people or expertise in a particular application domain. It does not evaluate the particular methodology used to develop the system. Instead it looks at the management and communication processes that an organization uses to ensure that the products built conform to requirements and that the techniques used to support the building of products are effective.

Continuous Process Improvement

Some of the key ideas of continuous process improvement for software that SCE team members should understand are summarized below. Refer to CMM v1.1 for a more in-depth discussion.

- The focus on software process as opposed to software technology is based on the principles of continuous process improvement and statistical quality control developed for the manufacture of hardware products.

  These principles were originally developed by Walter Shewart, W. Edwards Demming, Joseph Juran, and others for manufacturing applications. CMM v1.1 represents the application of process improvement principles of process improvement to software development.

- Continuous improvement in the software development process leads to improved software quality and to improved cost and schedule performance.

- In an organization with mature software processes, past performance is a more reliable indication of how well an organization might perform on a new development.

  Without processes that are defined and institutionalized across all projects, each project develops its own process. While such an organization may demonstrate good performance on some projects, performance on future projects is dependent on the individuals assigned to the new project.

- Process improvement is accomplished in discrete levels.

  Before process improvement can take place, processes must be defined and mechanisms must be in place to ensure that the processes are followed. The components of a lower level must be in place before an organization will receive the benefit from implementing a component of a higher level.

- Process improvement takes time.

  It takes a development organization a significant amount of time to make an improvement in their software process capability. Looking at the software processes used on current projects is a good indication of what an organization will be likely to achieve on the next project.

The Evaluated Organization

Many software development organizations are made up of a number of smaller organization units which operate with some degree of independence. These smaller organization units may work with different application domains or product types. Frequently an organization may be distributed among different physical locations. The software processes used on a given project may depend on which part of the organization will perform the work.

The SCE method is designed to evaluate the software process capability of that part of an organization that has primary responsibility for meeting the SCE sponsor's need with respect to developing and maintaining software products. The evaluation should focus on the site and on

the specific projects that will be most representative of that part of the organization which will be supporting the sponsor.

Process improvement activities may be carried out at different levels throughout the organization. Individual projects may be at different stages of implementation of standard practices. The SCE should focus on the continuous process improvement that is directly related to the work which will be done for the sponsor of the SCE.

Section 2   CMM-based SCE Data Collection Model

The current SCE method (Version 2.0) uses CMM v1.1 as the basis for investigating and making judgements about a development organization's software processes. In addition to CMM v1.1, the SCE Method uses materials derived from CMM v1.1 as part of the common rating framework effort at the Software Engineering Institute.

Collectively, these materials provide a rich information structure that is diagrammed (on a high level) in Figure 2-1 below. As indicated, the structure is not strictly hierarchical. Each Key Process Area (KPA) contains goals/subprocess areas which in turn contain key practices

Maturity Level
A well-defined plateau of process capability.

Key Process Area (KPA)
A cluster of activities that achieve a set of goals.

Goal /Subprocess Area
A focused subset of process activities that work toward achieving a specific KPA goal.

Feature
One of a set of attributes that indicate the implementation status of a KPA.

Key Practice
Infrastructure and activities that contribute most to the achievement of the KPA goals

Figure 2-1: CMM–based Data Collection Model

organized by features. However, each key practice may be associated with more than one goal/subprocess area. This section summarizes the structural components of the data collection model. More information is given in Appendix B.

Maturity Levels

CMM v1.1 provides a framework for organizing the evolutionary steps of process improvement into five maturity levels. A maturity level is a well-defined evolutionary plateau toward achieving a mature software process. Each maturity level provides a layer in the foundation for continuous process improvement. Maturity levels are listed in Appendix B, Section B.1 (➡page B-2).

Key Process Areas

Except for the initial level, each maturity level is characterized by several Key Process Areas (KPAs). Each KPA identifies a cluster of related activities that, when performed collectively, achieve a set of goals that significantly contribute to a level of software process capability.

The path to achieving the goals of a KPA may differ across projects based on differences in application domains or environments. Nevertheless, all of the goals of a KPA must be achieved for the organization to satisfy that KPA. When the goals of a KPA are accomplished on a continuing basis across projects, the organization can be said to have institutionalized the process capability characterized by the KPA. The KPAs are listed in Appendix B, Section B.2 (➡page B-3).

Goals

CMM v1.1 defines a set of goals for each KPA. The goals signify the scope, boundaries, and intent of each KPA. When evaluating a specific implementation of a KPA, the

goals can be used to determine whether an organization or project has effectively implemented a KPA. The KPA goals are listed in Appendix B, Section B.3 (➡page B-4).

Su   ss Areas

The goal statements in CMM v1.1 represent desired states that an organization's processes should try to achieve. However, what the team observes are the activities that are performed to achieve the goals. The SCE Method uses a defined set of subprocess areas to help teams identify these activities. The subprocess areas used in SCE were developed as part of the common rating framework development at the SEI.

A subprocess area is a set of activities in an implemented process that, acting together, attempts to achieve one of the goals of a KPA. There is a one-to-one correspondence between the subprocess areas and the KPA goals, and the subprocess area definitions are derived directly from the goal statements. The subprocess areas are listed in Appendix B, Section B.4 (➡page B-10).

Key Practices

Each KPA in CMM v1.1 is described in terms of the key practices that contribute to satisfying its goals. The key practices describe the infrastructure and activities that contribute most to the effective implementation and institutionalization of the KPA.

The key practices should be thought of as describing in high level terms "what" is to be done in all software development or maintenance scenarios. They should not be interpreted as mandating "how" the goals should be achieved. Many alternative methods of implementing the practices may accomplish the goals of the KPA for a particular development or maintenance scenario. The key practices should be interpreted rationally to judge whether the goals of the KPA are effectively, although perhaps differently, achieved.

Generally, the key practices are too low level to be used directly in an SCE evaluation. There are too many key practices for teams to try to investigate all of the individual practices. Also, if SCE teams were to work at that level of detail, the teams might tend to look for specific implementations of a practice rather than investigating the existing practices. However, the key practices are useful for adding context to the goals, and for giving SCE teams clues about what to look for during data collection. The key practices are described in *Key Practices of the Capability Maturity Model, Version 1.1* [Paulk 93b].

Features

Within CMM v1.1, the key practices for each KPA are organized by a set of *common features*. A common feature is "an attribute that indicates whether the implementation and institutionalization of a key practice is effective, repeatable, and lasting" [Paulk 93b]. The common features represent the necessary components of any process.

As part of the common rating framework development effort at the SEI, an expanded list of features was developed. The expanded list, referred to simply as *features*, is derived from the definitions and examples of the common features in CMM v1.1. The features are more appropriate for defining a single topic of investigation than the common features. The features used in SCE are listed in Appendix B, Section B.5 (➥page B-16).

Subprocess Area Matrix

SCE teams use a combination of features and subprocess areas to describe the topics to be investigated. One way to represent the way topics might be selected for investigation during an SCE is to visualize a matrix (or table) of subprocess areas and features. This provides a two–dimensional framework to help plan the investigation and organize the data collected. Each cell in the matrix defines

a potential topic for investigation. This concept is referred to as the *"subprocess area matrix"*, and is a component of the common rating framework being developed at the SEI.

Section 3   Summary of the SCE Method

SCE is based on the execution of a well-defined method by a team of trained experts. When used as part of a source selection process, provisions are present to insure fairness. For both source selection and contract monitoring applications the primary product of the SCE method is a set of findings about the evaluated organization's software process. The method requires that these findings be the consensus view of the evaluation team and that they be based on objective evidence. Each team executed phase of the SCE method (General Preparation, Specific Preparation, Data Collection, and Findings) is designed to support the goal of producing relevant and supportable findings.

Figure 2-2 (➥page 2-12) presents a top level diagram of the SCE process. *The activities within the outer dashed box* are part of the SCE method. The activities outside this box are necessary to support the use of SCE but they are not part of the method and are present to establish context. These activities will be determined by the intended use of the SCE.

The activities depicted within the inner dashed box are those normally performed by the SCE team itself. The activities shown outside this box are normally performed by the sponsoring organization.

Implementation of SCE for a Specific Application

The sponsoring agency must determine if the project is a good candidate for use of the SCE process. If it is determined that a SCE is appropriate, there are certain activities that the agency must perform before an SCE can be conducted. The activities are different depending on whether the SCE is to be used for source selection or contract monitoring (➥page 4-2).
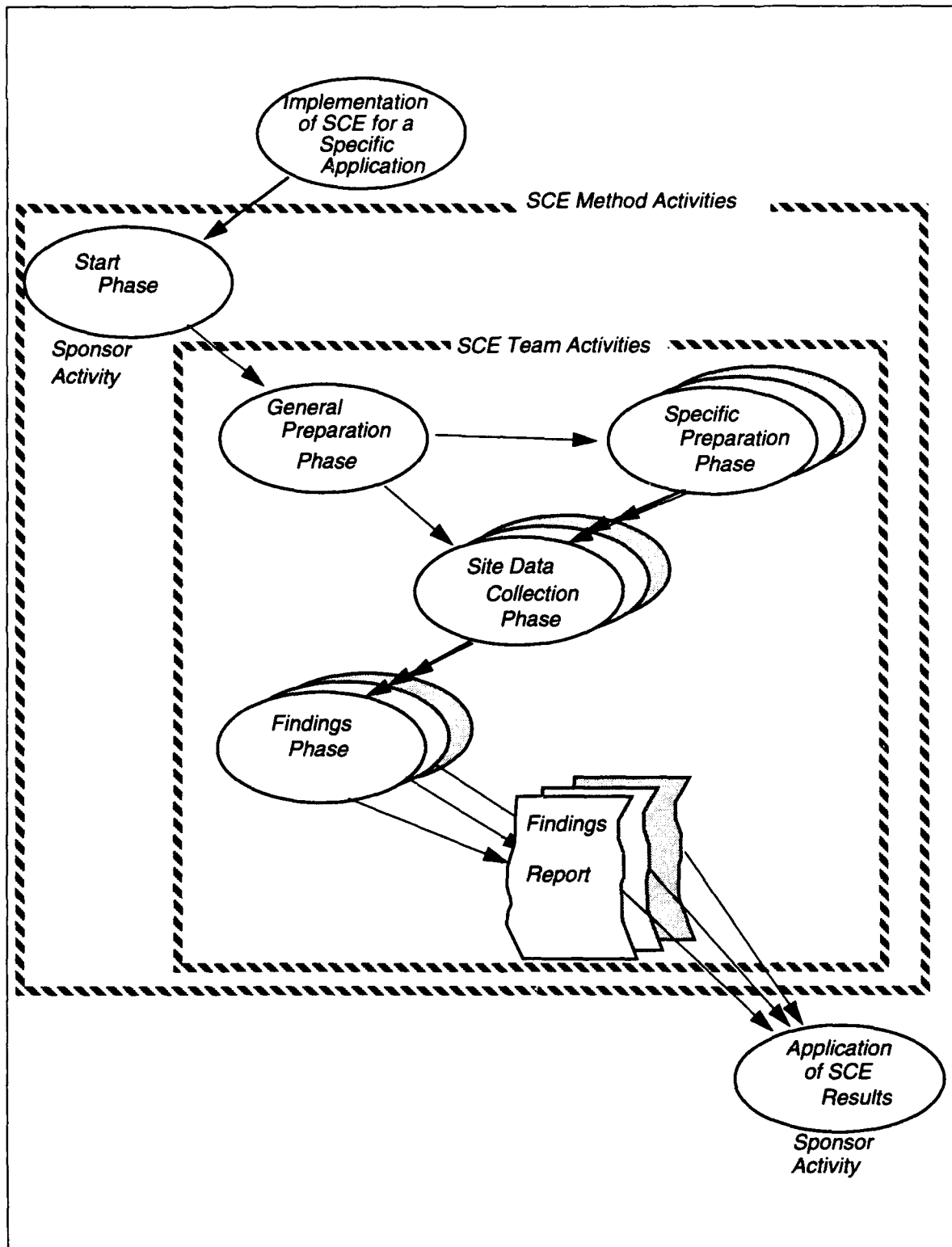
Figure 2-2: Overview of the SCE Process

For example, if SCE is used in a source selection, the evaluation plan must specify how the SCE results will be used. Appropriate items must be included in the procurement announcement and in the request for proposals. These items assure that offerors are aware of the fact that an SCE will be conducted, the process standard to be used, and the scope of what is to be evaluated. They also provide for submission of data to support the SCE process.

Refer to the *Software Capability Evaluation Version 2.0 Implementation Guide* [SCE 93a] for more detailed information on implementing SCE.

Phase 1: Evaluation Start

The Evaluation Start phase (➡page 4-14) is normally executed by the sponsoring agency. In this phase, they select the team to conduct the evaluation and provide the guidance to the team on what to evaluate.

The sponsoring agency describes the product associated with the sponsor's need in a standard format known as a *Target Product Profile* (➡page 4-15). The profile characterizes the desired product in terms of a set of standard attributes that allow the team to make effective comparisons among projects.

The sponsoring agency also specifies the *Target Process Capability* for this evaluation (➡page 4-16). The Target Process Capability lets the SCE team know which software development processes are viewed as most important for the sponsor's need. It is specified as a list of KPAs from CMM v1.1.

Phase 2: General Preparation

The General Preparation phase (➡page 5-2) is executed by the team in order to define the scope of the SCE investigated within the Target Process Capability. The primary output of this phase is a list of software

development activities that will be investigated at each of the development organizations evaluated. The activities to be investigated are specified as a subset of the subprocess areas associated with the goals for each KPA in the Target Process Capability. The subprocess areas selected as critical for evaluation are those most likely to reveal process related risk (➡page 5-13).

In addition to the information supplied by the sponsoring agency from the Evaluation Start phase, the SCE team receives information from the development organizations being evaluated. This information includes organization charts and profiles for the proposed project and for past and current projects that are candidates for evaluation (➡page 4-7).

When the SCE is used in source selection, the offerors' lack of experience on projects similar to the proposed development is a factor in selecting critical subprocess areas `N` n SCE is used in contract monitoring, critical subprocess areas may be selected based on an established process improvement plan or on other criteria specified by the sponsoring agency.

Both this phase and the Specific Preparation phase are used to develop the plans that will enable the team to use the limited on site time to focus on significant issues.

Phase 3: Specific Preparation

The Specific Preparation phase (➡page 5-29) is executed once for each development organization to be evaluated. Its purpose is to prepare a site visit plan to investigate the critical subprocess areas selected during General Preparation phase.

The team uses the information about the development organization from the General Preparation phase plus additional information such as responses to the *maturity questionnaire* (➡page 5-34) for the projects to be evaluated.

The team uses this information to determine the strategy for the investigation of each critical subprocess area at the particular site.

The plan will address people to interview, areas to be investigated, and documents to be reviewed. The plan, in the form of lists and worksheets, will be the starting point for a particular site visit that becomes the site data collection phase.

Phase 4: Site Data Collection

The Site Data Collection phase (➡page 6-11) is executed once for each development organization evaluated. Starting with the worksheets developed in the Specific Planning phase for the development organization, interviews are conducted and documents are reviewed. The team caucuses often to discuss the data they are getting, conclusions they are reaching, and other data they would like to have in order to understand the issues under investigation.

Activities are aimed at getting objective data to support conclusions about specific investigation topics. The data collection plan is revised as conclusions are reached and as progress is achieved. An important consideration is the limited amount of time available for the site visit. Focus on the essential data necessary to support conclusions must be maintained.

The concluding activity for this phase is to conduct a set of consolidation interviews. During these interviews, data is collected to obtain team consensus on open issues and to validate tentative team conclusions.

Phase 5: Findings

The final phase of the site visit is Findings (➡page 6-33). Findings document the team's consensus regarding the process capability of the development organization in the

Title

> *The finding*

Strengths

> *Observed stenghts relative to the finding*

Weaknesses

> *Observed weaknesses relative to the finding*

Improvement Activities

> *Observed improvement activitiesrelative to the finding*

Figure 2-3: Findings Format

subprocess areas investigated. There is a finding for each
Key Process Area investigated. Figure 2-3 presents the
general format of a findings sheet.

The title is the name of the KPA being discussed

Below the title there should be a clear statement of the
conclusion reached by the team regarding the Key Process
Area. It should be generally applicable to the organization
as defined in Section 1. The statement should support the

sponsor in determining the relative process risk in the subject areas across development organizations. This section should always be present. If a development organization has an acceptable process in the area investigated with no special strength, no weaknesses, and no improvement activities in progress then a finding of "Acceptable in this Key Process Area" is appropriate.

The strengths section is used to document process capability in excess of that which is needed to satisfy the KPA goals. If no strengths were found then this section should say "None observed."

The weakness section is used to document process capability that is less than the capability needed to satisfy the KPA goals. If no weaknesses were found then this section should say "None observed."

The Improvement Activities section is used to document organized efforts to improve the processes for this KPA. If no improvement activities were found then this section should say "None observed."

Application of SCE Results

The final activity depicted in Figure 2-2 () is the application of SCE results. This function is executed by the sponsoring activity and is the consumer of the SCE outputs. In a source selection, the findings are used to assess the process risk associated with each offeror to become an input to the source selection process.

Section 4   Benefits of Using SCE

SCE helps acquisition agencies evaluate the maturity of a development organization's software processes. This evaluation helps the acquisition organization in two ways:

- When used in source selection, SCE helps the acquisition agency to determine the risk associated with each development organization's software process maturity. The acquisition agency may use the risk as a factor in the selection of software contractors.

- When used for contract monitoring, SCE helps the acquisition agency identify the risk associated with a contractor's software process maturity and provides the acquisition agency with a method to assess the effectiveness of the contractor's process improvement activities.

Objectivity

For both source selection and for measuring process improvement (especially if tied to an award fee) there is value in assuring that the process used is objective. Key factors in the SCE method are evaluation relative to a public process capability standard (the CMM) and verification of findings about the process capability by means of collecting objective evidence. Further, in the source selection application, the method requires the same capabilities to be investigated with each offeror. These factors assure objectivity and also support comparison of results across offerors in source selection or across time in contract monitoring applications.

Realism

The SCE method focuses on actual processes being employed on current programs by offerors or contractors. Information is collected, correlated, and reported using specific engineering and/or management practice categories. Findings generated by the SCE team are based

on observable process artifacts and relate directly to current activity in specific capability areas. In the source selection application, comparison of current observable process capability (as documented by the SCE) provides the sponsoring agency with information by which to evaluate the realism of the proposed approach. In the contract monitoring application, current observable process capability (as documented by the SCE) can similarly be compared to that called for by the contractor's process improvement plan. This provides a realistic method of determining the value of the plan.

Motivation

Use of the SCE method in contract monitoring coupled with award fees tied to process improvement is a direct motivator. Use of SCE in source selection also has motivational power for long term process improvement. In particular, using SCE results as a discriminator in source selection ties success in new business acquisition directly to process improvement activity. As more development organizations implement process improvement programs and raise the level of their process capability, the government will have the option of setting higher standards and using contractors that have higher probabilities of achieving project cost, schedule, and quality goals.

Chapter 3   Team Development

Chapter 3   Team Development

This section provides background information to help a team develop the skills that are needed to conduct an SCE. This report assumes that the team has already been selected. The selection of the team is normally the responsibility of the organization that is sponsoring the SCE. Guidance for selecting team members may be found in the report *Software Capability Evaluation Version 2.0 Implementation Guide* [SCE 93a].

Section 1   Team Member Qualifications

The effectiveness of an SCE is highly dependent on the experience of the team members. The guidelines and heuristics provided in this document and in other SCE products can not substitute for adequate experience in software development and system acquisition. The method can provide a structure to direct how information is collected and organized. The supporting products can convey lessons learned from the experience of other teams and can point out areas that may be overlooked. However, the findings ultimately rely on the judgements of the team members as to whether the processes they observe are effective.

The *Implementation Guide* [SCE 93a] provides guidance on the qualifications for SCE team members. It describes the *recommended experience* levels that have been found to be effective.

Section 2   Training the SCE Team

SCE Team Training is a multi-day course that is offered by Software Engineering Institute for teams of experienced personnel who have been selected to conduct the SCE site visits. Completion of the *SCE Team Training* should be considered a requirement for all SCE team members.

The course provides instruction in how to conduct an SCE. The course includes guided and independent case studies which give team members a chance to practice preparing for and conducting an SCE using simulations of site visits. The course helps to develop the skills required to conduct SCEs effectively, and helps the group develop into a cohesive team.

Team members should also be familiar with the *Capability Maturity Model for Software, Version 1.1* [Paulk 93a] and *Key Practices of the Capability Maturity Model, Version 1.1* [Paulk 93b]. The CMM is the model used in the SCE method and it is also the model used by many organizations to plan their software process improvement activities.

Other references that may help the team members to prepare for conducting an SCE include:

- *Managing the Software Process [Humphrey 89]*

- *Software Capability Evaluation Version 2.0 Method Description* [SCE 93b]

- *Software Capability Evaluation Version 2.0 Implementation Guide* [SCE 93a]

Some organizations may be tempted to use this report and the references listed above to conduct an SCE without the formal *SCE Team Training*. This report was not designed for that purpose. The *SCE Team Training* provides the opportunity to participate in simulated site visits with feedback from experienced instructors. New teams can learn about problems that other teams have had. Without

this experience, teams are likely to be less effective. Many teams that have conducted SCEs have reported that they could have used more training and preparation.

Section 3   Team Building Activities

An essential aspect of preparing a team to conduct SCE site visits is performing team building activities prior to going on-site. There are a number of good references available on teamwork and team building. These topics are beyond the scope of this report. However, some issues of particular importance to conducting an SCE are discussed.

Being able to work effectively as a team is important to the success of an SCE in two ways. First, the team has a lot to accomplish in a short time during a site visit. The team members must be able to work efficiently as a unit to complete everything in the allocated time. Second, the team must be able to make decisions based on consensus. Judgements should be based on the combination of ideas and experiences of all members rather than on the more limited experience and potential biases of one or two strong personalities.

Team members, and especially the team leader, should make sure that they have the necessary skills to work together effectively as a team. In particular, they should develop teamwork and meeting management skills that will help them to:

- stay focused on the immediate task.

- ensure that everyone has a chance to express their opinions.

- move the discussion toward a consensus.

- recognize when continued discussion is inappropriate because additional information is needed or because the discussion is not pertinent to the current task.

SCE Team Training includes activities that will help to develop teamwork. However, many months may pass between when teams complete training and when they conduct site visits. There may be times when a team may

be made up of trained individuals who have not completed the training together. In these scenarios, team building is crucial, since the members have not operated as a team before. The team leader should consider including team building activities as part of the team's preparation, especially if the team members will not have an opportunity to work as a team on some significant tasks prior to the site visit.

An SCE team is neither an autocracy, where the leader dictates what decisions are made, nor a democracy, where the team votes and the majority prevails. Instead the decisions are reached by team consensus, meaning all members agree to the findings, and there is no significant minority dissent on issues. If consensus on an issue cannot be reached, then there is no finding in that area.

All team members should be aware of their role in the team decision making process. An effective team will have the following characteristics:

- All members participate. No member dominates or makes others hesitant to express an opinion.

- The working environment is comfortable and friendly. Communication is open. There are no hidden agendas. The team establishes a climate of trust.

- Team members maintain an open mind and try to understand the others point of view. They are active and effective listeners.

- Conflict is addressed openly and resolved through collaborative negotiation. The team is comfortable with disagreement and works to find a solution rather than avoiding or suppressing conflict.

The team building activities, whether conducted as separate exercises or as part of the normal work of the team, should be used to ensure that the individual members will be able to work effectively as a team. A site visit is not the place to find that a particular team member will disrupt the smooth operation of the team. Someone

with the best of intentions could make the team ineffective by arguing their own positions excessively or by demanding an unreasonable amount of evidence before agreeing to any decision. Also a person who never contributes to the discussion is of limited value. The decision process works best when judgements are based on the experience of all of the members.

Section 4   Team Leader

Experience shows that an effective team leader is critical to the operation of the team. The team leader is the SCE team focal point for both the acquisition office and the development organization. The team leader is responsible for planning and scheduling the SCE activities and for ensuring that the team meets its schedule and objectives. The leader should have enough basic leadership skills to ensure that the team functions effectively.

The team leader should be the one most qualified, based on knowledge, experience, and amount of direct SCE site visit experience. Every graduate of the SCE training program should be a member rather than a leader of an SCE Team for at least two evaluations. Junior- and mid-level personnel should take part in at least three SCEs before being considered for the team leadership role.

Section 5   Practice SCE

Individuals returning from their first SCE commonly report that they could have used more team training, preparation, and time to conduct the interviews. Some teams have conducted one or more practice SCEs on friendly organizations before the beginning the source selection. A team should consider the use of a practice SCE, especially if no one on the team has prior SCE experience.

The practice SCE will show the team members how well they understand the method and will give them a chance to exercise the mechanics of the interviewing, document review, and findings preparation activities, and to review any problem areas before conducting the actual SCE. It will also give them valuable interviewing experience that will show them the types of questions that are effective and those that are not. The organization evaluated will also benefit by getting the feedback on their software development process.

Practice SCEs should be conducted within the team's own agency or with another government agency. Do not use an organization that the team may have to evaluate in the current acquisition or in some future acquisition. This could give them an unfair advantage over the other offerors.

Section 6   Legal Briefing

SCE team members need to be aware of the legal issues involved when using SCE in either a source selection or contract monitoring role. There are constraints on what a team member can say about the product being acquired or about how the SCE results will be used. The team is also restricted in how it will conduct the investigation so that all development organizations are evaluated fairly.

The legal constraints on team members are primarily defined in the Federal Acquisition Regulations but each branch of the government may have additional policies or requirements that must be followed. The team members should receive a legal briefing that clearly defines the constraints that must be followed for the specific acquisition. The briefing should be conducted before the team has any contact with the development organization.

Part 2   Instructions and Guidelines

Chapter 4   Before the SCE Team Begins

Chapter 4   Before the SCE Team Begins

As a team member, you will normally begin your activities with Phase 2 of the SCE method: General Preparation. However, there is some preparation that must be completed first in order to use SCE legally and effectively for a specific application. These preparations are normally made by the sponsoring agency. They are part of the implementation of the SCE.

This chapter provides a summary of the SCE preliminary implementation activities to establish the context for the SCE team. Even though you may not be directly involved in these activities, you should understand what they are and why they are important. Also, you may want to verify that these preparations have been completed since they may affect the success of the evaluation.

In some cases, the SCE team leader or other team members may be asked to perform or participate in these activities but this is beyond the normal team responsibilities and outside the scope of this document. If you will be participating in these activities, refer to the *Software Capability Evaluation Version 2.0 Implementation Guide* [SCE 93a] for more detailed information.

This chapter is divided into three sections:

- Section 1 describes the preparations that must be made to use the SCE method for a given application.

- Section 2 describes the information which the team will need from the organizations to be evaluated.

- Section 3 describes the steps of Phase 1 of the SCE method: Evaluation Start.

Section 1   Using the SCE Method for a Given Application

The sponsoring agency must make certain preparations in order to use the SCE method legally and effectively in the acquisition process. These preparations are different depending on whether SCE will be used for source selection or for contract monitoring.

This section provides a brief description of the preparation activities to make team members aware of the issues and to show where the team's activities fit in the overall process. More detailed guidance for implementing SCE in a source selection is provided in *Software Capability Evaluation Version 2.0 Implementation Guide* [SCE 93a]. Guidance for implementing SCE for contract monitoring will be included in future versions of the *Implementation Guide*.

Source Selection

When deciding to use SCE as part of the source selection process, the sponsoring agency must first decide whether it will be cost effective based on the significance of the procurement. The Acquisition Plan should include guidance to determine whether an SCE should be performed, and if so, how the results are to be used. See the *Implementation Guide* for recommended criteria to make this determination.

Assuming that the decision is made to conduct an SCE, the acquisition announcements in the Commerce Business Daily should include a statement that an SCE will be conducted. This will let potential offerors know up front that they will be expected to support an SCE. It will also give organizations that may not have gone through an SCE before a chance to learn what ᵗ   ᵉct.

The Source Selection Plan should define how SCE will be use in the source selection process. It should define the role of the SCE team relative to the other key personnel in the source selection process. If there is not a separate supporting document in the form of a Source Selection Evaluation Plan, the Source Selection Plan should also describe how the results of the SCE will be applied and integrated into the source selection process.

The offerors should be briefed at the pre-proposal conference on what to expect when an SCE is conducted. The briefing should explain what an SCE is, how the evaluation will be conducted, the types of documentation the team may look at, and the type of people that may be interviewed. The Briefing should also include a sample site visit schedule.

The Request for Proposals must explain how SCE will be used as an evaluation criterion and identify what will be evaluated. It should also include instructions for the information that the offeror must supply for the SCE team to use in preparation for the site visits.

The Source Selection Evaluation Plan must specify how the SCE findings will be evaluated and how much weight will be given to the SCE findings relative to the other evaluation criteria. The Source Selection Evaluation Plan must be prepared before the proposals are received.

The following checklist may be used to ensure that the necessary preparations have been done to use SCE in a source selection. For more details on each activity refer to the *Software Capability Evaluation Version 2.0 Implementation Guide* [SCE 93a].

Source Selection
Implementation
Checklist

❏ Acquisition Plan includes guidance to determine if SCE should be performed.

❏ Acquisition Announcements include a statement that an SCE will be conducted.

❏ Source Selection plan describes how the results of the SCE will be applied and integrated into the source selection process.

❏ Pre-proposal conference includes a briefing on what to expect when an SCE is conducted.

❏ SCE is incorporated in the evaluation plan.

❏ The Request for Proposals delineates SCE as a criterion and includes instructions for supplying the information required to prepare for an SCE.

❏ Proposals are received and evaluation is completed on schedule.

❏ SCE Information to be provided by the offerors has been received by the team.

Contract Monitoring   When deciding to use SCE for contract monitoring, the sponsoring agency must first decide whether it will be cost effective based on the significance of the contract. The sponsoring agency must determine how SCE will be integrated with other contract monitoring activities. An SCE may be conducted after contract award to identify potential risks with the processes that are being applied and executed on the contract. It may also be used to monitor process improvement over the course of the contract.

When SCE is used to monitor process improvement, a "baseline" of process capability is established. Interim evaluations are planned to monitor improvements in the processes and to monitor concurrent increase in capability over the life of the contract. In this environment the SCE concentrates on the processes of one development organization for the project under contract.

The sponsoring agency must determine how frequently SCEs will be conducted and how the results will be used. When SCE is being used to monitor process improvement, the sponsoring agency and the development organization should have a documented process improvement plan that identifies the areas of concern and the priority for making improvements. This will allow the SCE team and the personnel assigned for monitoring of the SCE results to focus on the areas that are the most significant.

The contract should specify when SCEs will be conducted and how the results will be used. Where appropriate, the Request for Proposals should specify that SCE will be used for contract monitoring so that the cost of conducting the SCEs can be negotiated competitively. If the use of SCE is not included in the awarded contract then a change must be negotiated.

The development organization will need to understand what SCE is, what they will need to do to support the SCE, and how the results will be used.

The information that the SCE team will need for preparation must be requested in advance. The information must be received before the team starts the preparation activities.

The following checklist may be used to ensure that the necessary preparations have been completed to implement SCE for contract monitoring.

Contract Monitoring
Implementation
Checklist

❑ The sponsoring agency has determined how frequently the SCEs will be conducted and how the results will be used

❑ The development organization has been briefed on SCE

❑ The sponsoring agency and development organization have agreed to and documented a plan for software process improvement

❑ The contract or contract modification specifies how SCEs will be conducted and how the results will be used

❑ The information that the development organization must supply has been requested

❑ Information to be provided by the development organization has been received

Section 2   **Requesting Information from the Development Organizations**

The SCE team needs the following information from the development organizations being evaluated:

- A Proposed Project Profile,

- Project Profiles for each of the projects which are offered for evaluation,

- Organization charts, and

- Responses to the Maturity Questionnaire from each of the projects which are offered for evaluation.

Before the team can begin the preparations for any of the site visits, they need all of the information from each of the organizations to be evaluated except for the responses to the Maturity Questionnaire.

In a source selection, the information is normally requested as part of the offeror's proposal package. The sponsoring agency will specify the information needed in the Request for Proposals.

In a contract monitoring application, the contract or contract modification should specify the information needed.

The *Software Capability Evaluation Version 2.0 Implementation Guide* [SCE 93a] contains suggested instructions that may be used to request the information.

Instructions   **Requesting a Proposed Project Profile**

The Proposed Project Profile communicates the development organization's view of the product and of the project required to produce it. It is used by the SCE team to determine the types of experience required for the project and to help select projects for evaluation.

The *Implementation Guide* contains sample instructions that may be used to request the Proposed Project Profile. The development organization will need the following information to complete the forms:

❏ Blank Proposed Project Profile forms	Appendix E
❏ Attribute Definitions	Appendix C

You may need to provide some additional guidance on how to determine the project attributes so that they are consistent for all organizations and projects. Refer to "Guidelines, Determining Project Attributes" (➡page 4-12) for more information.

Instructions   **Requesting Project Profiles**

The Project Profiles provide information about the development organization's projects that are being offered for evaluation. It is used when multiple projects will be evaluated for an organization such as when SCE is used in a source selection application.

The information allows the SCE team to compare the experience that the organization has gained from those projects with the experience needed for the new development. It is used to identify areas where a lack of experience may increase risk. The information is also used to help select projects to investigate and to determine what is to be investigated.

The *Implementation Guide* contains sample instructions that may be used to request the Project Profiles. The development organization will need the following information to complete the forms:

❏ Blank Project Profile forms	Appendix E
❏ Attribute Definitions	Appendix C

Additional guidance on how to determine the project attributes so that they are consistent for all organizations and projects is provided in "Guidelines, Determining Project Attributes" (➡page 4-12).

Instructions   **Requesting Organization Charts**

The organization charts are used by the SCE team to help plan the interview strategy for the site visit. They may also be used as a source of information for selecting subprocess areas to evaluate and for developing topic lists.

The organization charts are usually provided in the company's standard format. Some general guidance should be provided to the development organization to make sure that the necessary information is included:

☞ Show the organization of each project submitted for evaluation.

☞ Show how the projects are overseen by upper management.

☞ If the projects are from more than one organization, show how the organizations are related.

☞ Show how the engineering departments (software engineering, testing, quality assurance, etc.) relate to the various projects and to upper management.

Instructions   **Requesting Questionnaire Responses**

The Maturity Questionnaire is used by the SCE team to gain initial insight into the development organization's processes. It is a potential source of issues to investigate during the site visit.

[This section will be completed when the new MQ and instructions are available.]

Guidelines   **Determining Project Attributes**

Appendix C contains the definitions for the project attributes used in the SCE method. These definitions should be provided to the development organizations to help them complete the profiles.

For some of the attributes, additional clarification may be required so that all of the organizations use the attribute consistently. There is no standard industry convention for specifying some of the attributes such as application domain, product type, and project size. Additional guidance should be provided to the development organizations on how to specify those attributes so that comparisons will be meaningful.

Application Domain and
Product Type

Application domain and product type are both widely used terms but the meaning can vary greatly. For example, some people may think of an application domain as a general class of system types such as real-time, embedded, or MIS. In the SCE context application domain and product type refer to something more specific. It indicates the areas of knowledge that are needed to understand the requirements for the system.

One way to communicate the level of information desired is to provide a list of application domains and product types as examples. The list should include a description of the characteristics that distinguish the application domain or product type and identify the types of knowledge or skills associated with the domain. Make it clear that the list provided is not meant to be all inclusive.

Project Size

The software team size portion of the project size attribute refers to the maximum software engineering team size. The software engineering team includes those people

directly involved with design, code, and integration. Management and support groups such as CM or QA are not included.

Code size should include all developed code. The percentage or size of new, modified, reused, and undelivered code should be enumerated. The guidance should specify:

- Whether to count statements or lines of text.

- Whether to count comments.

- Whether to count data declarations.

Section 3   Evaluation Start (Phase 1)

Objective   To select and train the team to perform the evaluation and
                     to provide guidance to the team about the product to be
                     developed and the process areas of concern.

Participants   Sponsoring Agency

Time   2-4 months (primarily due to the time required to schedule
                     training for the SCE team. If the team is inexperienced in
                     the SCE method, additional time should be allowed for a
                     practice SCE)

Entry Criteria

❏ The decision has been made to use the SCE method.
❏ The product to be built has been defined.

Activities

❏ Develop Target Product Profile (Step 1)	page 4-15
❏ Determine Target Process Capability (Step 2)	page 4-16
❏ Select SCE Team (Step 3)	page 4-18

Step 1   Develop Target Product Profile

Objective

Understand the attributes of the software product and the project required to produce it.

Description

The first step in the SCE method is to prepare a Target Product Profile for the product being built. Preparing the Target Product Profile causes the sponsoring agency to think about the characteristics of the product that might affect the software development processes needed. It also helps to communicate the sponsoring agencies understanding of the product to the SCE team.

The Target Product Profile describes the product in terms of a standard set of attributes. The standard attributes provide a convenient way make comparisons with other projects to assess the experience the development organizations and the end users have that is applicable to the new project.

The attributes also serve as a checklist that helps to ensure that all of the important characteristics of the project are considered.

Appendix C contains the definitions for the attributes used in the Target Product Profile.

Step 2   Determine Target Process Capability

Objective

Determine the process capability that is most appropriate for the planned development - the Target Process Capability.

Description

The Target Process Capability for an SCE is defined as the discrete subset of the CMM KPAs that development organizations are evaluated against, and describes the capabilities that the acquisition agency seeks in the supplier for this contract.

| Important! | The Target Process Capability determines which KPAs the SCE team will investigate. It does not specify the process capability maturity level that an organization must have to qualify for the contract. |

The recommended approach is to use *defined* as the Target Process Capability and to use all KPAs within the *repeatable* and *defined* maturity levels.

The *repeatable* level may be used as the Target Process Capability if a higher level capability is not required for the particular procurement. In that case, all of the KPAs from the *repeatable* level should be used.

For whatever Target Process Capability is selected, all KPAs from that level and all lower level KPAs should be used.

Because process maturity is built in stages, with the KPAs serving as building blocks of capability, mixing and matching of KPAs from within levels is not recommended. The KPAs within a level depend on each other. For example, in the *defined* level, the effectiveness of an organization's process definition activities may depend on

the organization process focus. Similarly, peer reviews depend on the organization's process definition and its training program.

Step 3   Select SCE Team

Objective   Have a trained team in place to execute the SCE.

Description   The sponsoring agency must select the team of 4 to 6 people
                       to conduct the SCE. This team will carry out the remaining
                       steps of the SCE method. The team may actually be
                       selected before Steps 1 and 2 are completed. In that case,
                       the team may perform or assist with those steps as well.

                       When SCE is used as part of the source selection process
                       the same team is used to conduct all of the evaluations.
                       This is necessary in order to ensure fairness in the selection
                       process.

                       The *Software Capability Evaluation Version 2.0
                       Implementation Guide* [SCE 93a] contains information
                       about the recommended experience levels and
                       qualifications for team members. Refer to that document
                       for more detailed guidance on selecting team members.

                       It is important to allow adequate time between selecting
                       team members and scheduling the site visit to allow for
                       training, team building activities, and preparation
                       activities.

Chapter 5   Preparing for the Site Visit

Chapter 5   Preparing for the Site Visit

This chapter provides detailed instructions for the tasks that SCE team members perform to prepare for a site visit. It covers Step 4 through Step 11 of the SCE method. These steps make up the General Preparation Phase (Phase 2) and the Specific Preparation Phase (Phase 3) of the method. Step 1 through Step 3 are usually completed before the team is assembled. Those steps are discussed in Chapter 4, Section 3 ().

Section 1   General Preparation (Phase 2)

Objective

Complete the portion of the site visit preparations that applies to all of the development organizations.

Participants

SCE Team Members

Time

1-2 days

Entry Criteria

❏ The steps of Phase 1 have been completed. The following outputs of Phase 1 are used in this phase:	page 4-14
❏ Target Product Profile	page 4-15
❏ Target Process Capability	page 4-16
❏ The SCE team has received the information requested from the development organizations. The following information is used in this phase:	page 4-7
❏ Proposed Project Profiles	page 4-8
❏ Project Profiles	page 4-9
❏ Organization Charts	page 4-10

Activities

❏ Create Experience Table (Step 4)	page 5-3
❏ Create Critical Subprocess Area List (Step 5)	page 5-13
❏ Originate Validation Worksheets (Step 6)	page 5-26

Step 4   Create Experience Table

Objective

Identify areas where of the development organizations lack experience, indicating a potential for risk.

Description

The SCE team does not have time to look at all of a development organization's software processes during a site visit. They need to focus on processes that are most important for the new development effort. One method used to select processes to investigate is to look at whether the development organizations have experience with similar projects. If the new project is significantly different from anything they have done in the past, then certain processes will be required to manage the risk involved with doing something new.

In this step, the SCE team evaluates the experience of each of the development organizations relative to the planned development by:

1. Preparing a Mismatch Identification Table to record the experience mismatches for each development organization.

2. Preparing an Experience Table that summarizes the relevant experience for all organizations.

The Experience Table is used in Step 5 (➡page 5-13) to help select subprocess areas to investigate during the evaluation. The Mismatch Identification Tables are used primarily for developing the Experience Table but they may also be used in Step 7 (➡page 5-31), to help in selecting projects to evaluate, and in Step 9 (➡page 5-39), to help develop investigation topics.

The topics in this document most applicable to this step are listed below.

Chapter 5	Guidelines, How to Determine Experience Mismatches	page 5-5
	Instructions, How to Prepare a Mismatch Identification Table	page 5-9
	Instructions, How to Prepare an Experience Table	page 5-11

Guidelines   **How to Determine Experience Mismatches**

The SCE team evaluates the relevant experience of a development organization by comparing the Project Profiles and the Proposed Project Profile that the organization submitted (➥page 4-7). The Proposed Project Profile represents the development organization's view of the planned development. The Project Profiles describe the projects that the development organization submitted for evaluation.

The profiles describe the projects in terms of a standard set of attributes. The attributes provide a convenient way to compare important characteristics of the projects. Appendix C contains the definitions of the attributes used in the SCE method.

The following guidelines for determining experience mismatches are provided.

General Guidelines   ☞ The comparisons look for similarities, not for exact matches.

                                ☞ The SCE method requires that a mismatch be obvious. If the team does not have consensus, then there is no mismatch.

Evaluating Application   ☞ Since there is no standard convention for describing
Domain and Product   application domain and product type, the responses
Type Attributes   may vary widely. These attributes should not be
                                   considered a mismatch just because the development
                                   organization used different terms.

                                ☞ Determine the areas of expertise that are important to this development and to what extent the application domain for the project provides that type of experience. Consider such things as:

- Does the new project require familiarity with a particular area of scientific knowledge or of a particular area of business?

  For example, radar systems and active sonar systems may be very similar in principle but the physical laws governing the propagation of electromagnetic waves through air and the propagation of sound through water are very different.

- Will the requirements be expressed using terms that are unique to the application domain or that have a special meaning in that domain?

- Do the systems that the product will be working with have interface characteristics that are unique to that application domain?

- Does the application domain or product type require different development methodologies?

  For example the methodologies used to develop management information systems are different from the methodologies used to develop real-time systems.

- Does the application domain or product type indicate a different operating environment?

  For example, shipboard systems have different environmental requirements than air-born systems.

- Is a different level of reliability required?

  For example, the level of reliability required for a life support system is much greater than that required for a management information system.

Evaluating Product Size Attributes

☞ Evaluate each of the numbers separately. A significant difference in any one of the numbers should be treated as a mismatch.

☞ Look for differences that are large enough that they would affect the process.

- Will the increase in software team size require the development organization to go from a single team to multiple teams working in cooperation (e.g., software team increases from 30 to 60 or the number of levels of management increases)?

- Will the organization require different procedures to staff up and train for the new project (e.g., staff size is 50% larger than current projects)?

- Is the schedule compressed so much that tracking and corrective action processes may not be adequate (e.g., going to a schedule half as long as current projects)?

- Is the schedule so long that turnover rates will likely be more of a factor than on current projects (e.g., going from 1-3 year projects to a 5 year project)?

- Is the code size large enough that current design and documentation methods may not be adequate or that current facilities, tools, and processes may not handle the throughput (e.g., code size increases by an order of magnitude)?

☞ The organization charts submitted by the development organization may be another useful source of information for comparing project size. If the organization for the proposed project is significantly different from that of current projects, consider whether the change could be caused by project size.

Evaluating Type of Work Attributes

☞ When evaluating the type of work on previous projects look for such things as:

- Did the development organization perform the same tasks as those required for this procurement?

- Did the development organization have the same scope of control for the project activities?

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 4   Create Experience Table
Guidelines   How to Determine Exp⸱⸱⸱ce Mismatches

For example, if the development organization's
previous projects have all been full development
then the development organization may not have
processes that will support coordinating work
among several organizations.

• Did the development organization have the same
  role in defining the requirements for the system?

  For example, in a full development the
  development organization may work with the
  acquisition agenc   ⸱efine the requirements
  while in a delive   : . type contract the
  acquisition agency n ⸱ provide the
  requirements.

• Does the acquisition agency have the same level
  of visibility and control?

**Evaluating
Subcontractors
Attributes**

☞ If the development organization does not plan to use
  subcontractors on the new project then whether
  subcontractors were used on other projects is not
  significant.

☞ If subcontractors will be used on the new development,
  then it is not only important that subcontractors were
  used on other projects but that the they were used in
  similar roles or with a similar level of responsibility.

For example the processes that are required to select
and to monitor a subcontractor that will develop a
critical portion of new software that must be integrated
by the prime development organization are different
from those required for subcontracting with a particular
vendor to retarget their software for the new
environment.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 4   Create Experience Table
Instructions How to Prepare a Mismatch Identification Table

Instructions

How to Prepare a Mismatch Identification Table

A Mismatch Identification Table is created for each development organization to be evaluated. The primary inputs are the Proposed Project Profile and the Project Profiles submitted by the development organization (➥page 4-7).

To prepare a Mismatch Identification Table:

1. Make a copy of a blank Mismatch Identification Table form from Appendix E.

2. Enter the name of each project at the top of a column in the row labeled **Projects**.

3. For each project, compare each of the attributes listed on the Project Profile with the corresponding attributes on the Proposed Project Profile. Refer to "Guidelines, How to Determine Experience Mismatches" (➥page 5-5) for more information. Place a "1" in the Mismatch Identification Table when the attributes match and a "0" when there is a mismatch.

4. For each row of the attributes, if a "0" is entered in all of the project columns, enter the abbreviation of the attribute in the **Result** column. If there is a "1" in any column of the row (i.e., if there is previous experience) then the **Result** column is left blank.

**Abbreviations for
Attribute Names**

• "ApD" for application domain	• "Lg" for Language
• "Pt" for Product Type	• "Tg" for Target
• "Tw" for Type of Work	• "Std" for Applicable Standards
• "Ps" for Project Size	• "Cmr" for Customer
• "Sc" for Subcontracting	

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 4   Create Experience Table
Instructions How to Prepare a Mismatch Identification Table

1 Make a copy of a blank Mismatch Identification Table form from Appendix F

2 Enter the name of each project in this row of headings

Mismatch Identification Table							
Projects	*Able*	*Baker*	*Charlie*	*Delta*	*Enigma*	*Fiesta*	**Result**
Major Attributes							
Application Domain	0	0	1	0	0	0	
Product Type	1	1	1	0	0	0	
Product Size	0	0	0	0	0	0	*Ps*
Type of Work	1	1	0	1	1	0	
Subcontractors	1	1	1	1	1	1	
Minor Attributes							
Language(s)	0	1	1	0	0	0	
Target(s)	0	0	1	0	0	1	
Applicable Standards	0	1	1	0	0	0	
Customer	1	1	1	0	1	1	

0 = experience mismatch, 1 = experience match

3 These columns are used to record the results of comparing the Proposed Project Profile to the Project Profiles for individual projects. Enter a "1" when the attributes match and a "0" when there is a mismatch.

4 The last column is used to record the development organization's combined experience. For each row, if all of the entries in the row contain "0", enter the abbreviation for the attribute. If any entry in the row contains a "1", leave the field blank.

Figure 5-1: Preparing a Mismatch Identification Table

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 4   Create Experience Table
Instructions How to Prepare an Experience Table

Instructions   **How to Prepare an Experience Table**

The SCE team creates one Experience Table for all development organizations in an evaluation. The inputs are the Mismatch Identification Tables prepared for each specific development organization (➡page 5-9).

To prepare an Experience Table:

1. Make a copy of a blank Experience Table form from Appendix E.

2. Enter the name of each development organization at the top of a column under the heading **Offerors.**

3. For each development organization, copy the values from the **Result** column of the Mismatch Identification Table created for the organization to the column in the Experience Table under that organization's name.

4. For each row of attributes, if the abbreviation is entered for any of the development organizations, enter the abbreviation in the **Result** column. Otherwise, the **Result** column is left blank.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 4   Create Experience Table
Instructions   How to Prepare an Experience Table

1 Make a copy of a blank Experience Table form from Appendix F

2 Enter the name of each development organization in this row of headings

Experience Table

Attribute Name	Offerors			
	Sigma Tech	*Beverly Ind*	*Crystal City*	**Result**
Major Attributes				
Application Domain				
Product Type		*Pt*		*Pt*
Product Size	*Ps*	*Ps*	*Ps*	*Ps*
Type of Work				
Subcontractors				
Minor Attributes				
Language(s)				
Target(s)				
Applicable Standards		*Stds*	*Stds*	*Stds*
Customer				

3 Copy the entries from the **Result** column of the Mismatch Identification Table created for each organization into these columns

4 For each row of attributes, if there is an entry for any of the development organizations, enter the abbreviation in the **Result** column. Otherwise, the **Result** column is left blank.

Figure 5-2: Preparing an Experience Table

Step 5   Create Critical Subprocess Area List

Objective

Define and document the scope of the SCE in terms of critical subprocess areas within the Target Process Capability KPAs.

A subprocess area is a process activity or group of closely related process activities. There is one subprocess area for each KPA goal. Each subprocess area focuses on a particular part of the KPA activities that are necessary for achieving the goal.

The topics in this document most applicable to this step are listed below.

Chapter 5	Guidelines, Selecting Critical Subprocess Areas	page 5-17
	Instructions, How to Select Critical Subprocess Areas	page 5-18
	Guidelines, Importance of Individual Project Attributes in Determining Risk	page 5-20
	Instructions, How to Prepare a Key Issue Table	page 5-23

In the limited time allotted to a site visit, it is not possible to look at all of a development organization's software processes in detail. Subprocess areas allow the SCE teams to focus on the particular part of an implementation of a process that relates directly to the KPAs

The critical subprocess areas are the set of subprocess areas selected by the SCE team for evaluation. The same set of critical subprocess areas is used for all development organizations being evaluated.

In this step, the SCE Team:

1. Selects the critical subprocess areas to be investigated.

2. Records the critical subprocess areas on the Key Issue Table.

The SCE team considers the following when selecting critical subprocess areas:

- What are the basic processes that a development organization would need for any software development effort?

- What processes would an organization need to manage for aspects of the project that are new to the organization?

- If the product being developed is new to the end user, what processes will the development organization need to manage the anticipated requirements changes?

- Are there any other considerations for the planned development that affect what software processes are needed?

Nucleus Software Development Capability

Figure 5-3 is a simple representation of why different levels of process capability might be needed for different projects. Consider an organization that has developed a series of products represented by X, X', and X''. If the products are basically the same, then a development organization will need certain processes to manage the development effort. These processes are referred to as a nucleus capability. They are the basic set of processes that a development organization would need to manage any software project.

Development Organization's Lack of Experience

Suppose the same organization decides to take on a different type of project represented by Y. If Y is significantly different from X, X', and X'' then there will be certain risks associated with those differences. The development organization will need to make changes to the way the processes are implemented to manage those risks. The specific processes needed will depend on what the new

aspects of the project are and on how great the difference is. Some of the processes needed to manage the risk may be the same as processes that are part of the nucleus capability but now they have an additional significance.

Operational Precedence   Another factor that can impact what processes a development organization needs is the amount of experience the end user has with similar systems. This is referred to as operational precedence. If the end user has not used similar systems or if the new system will be used in a different way, they will be less likely to know exactly what their needs are when the requirements are initially defined. The volume of requirements changes will probably be greater than would normally be expected. The development organization will need processes to help them manage those requirements changes.
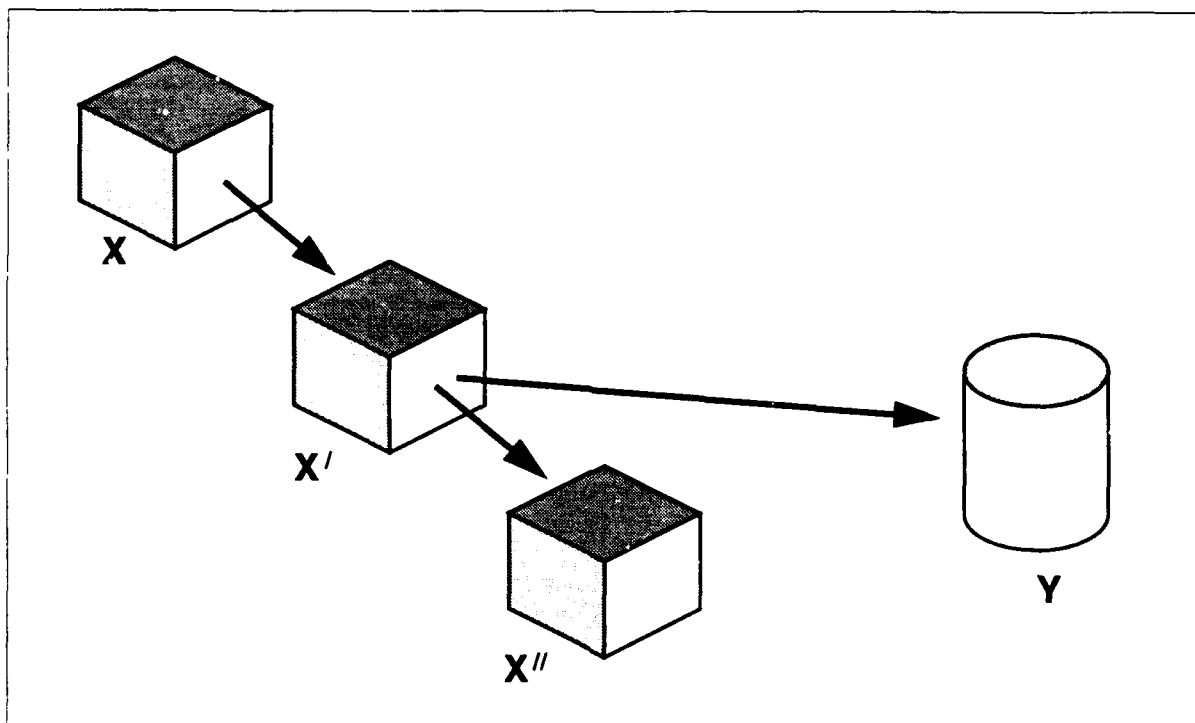
Figure 5-3: Process Capability Requirements

Other Considerations   The SCE team may know about other factors that may
                        affect the processes needed for the planned development.
                        For example, the project may require new technology that
                        has never been used in this type of application.

Guidelines   **Selecting Critical Subprocess Areas**

Critical subprocess areas establish the criteria that all development organizations will be evaluated against.

☞ There must be at least one subprocess area selected for each KPA in the Target Process Capability.

☞ The team should select from 13 to 20 critical subprocess areas for evaluation.

☞ To ensure fairness in a source selection application, all development organizations must be evaluated against the same set of subprocess areas.

Instructions

How to Select Critical Subprocess Areas

To select the list of subprocess areas do the following:

1. Construct a preliminary list by performing a table look-up, using the Subprocess Area Selection Tables in Appendix F, Section F2 (➡page F-5). Select only subprocess areas for the KPAs that are part of the Target Process Capability.

   a. The **Result** column of the Experience Table created in Step 4 (➡page 5-3) lists the major attributes of the new project for which any of the development organizations may have a lack of experience. Make a list of subprocess areas that are associated with these attributes in the tables.

   b. The Operational Precedence attribute is listed on the Target Product Profile created in Step 1 (➡page 4-15). If there is a lack of operational precedence for the system to be built, add to the list the names of those subprocess areas that are associated with the operational precedence attribute in the tables.

   c. Add to the list the names of those subprocess areas that are associated with a nucleus capability in the tables.

   Appendix F, Section F1 (➡page F-3) contains a table for selecting critical subprocess areas based on the size of the development undertaking. This table may be used instead of tables in Section F2 in cases where there are no experience mismatches. It may also be used in conjunction with the tables in Section F2 either to indicate additional subprocess areas to look at or to help reduce the set of subprocess areas selected if the initial list is too large.

2. Select additional KPAs based on the SCE team's knowledge of other factors that may affect the processes needed for the planned development.

3. Reduce the list to a reasonable number of areas to investigate during the site visit. It might be helpful to prioritize the subprocess areas within each KPA. Then select the one or two highest priority subprocess areas from each KPA.

4. Review the list to make sure that:

   a. There is at least one subprocess area for each KPA in the Target Process Capability.

   b. All subprocess areas listed are for KPAs in the Target Process Capability.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 5   Create Critical Subprocess Area List
Guidelines   Importance of Individual Project Attributes in Determining Risk

Guidelines

Importance of Individual Project Attributes in Determining Risk

The Subprocess Areas Selection Tables in Appendix F indicate subprocess areas that might be important if an organization lacks experience for each of the major attributes. The following provides guidance on understanding how the attributes and subprocess areas are related.

Application Domain and Product Type

The application domain and product type determine the subject matter expertise that is necessary to translate system requirements into software requirements and into the software top level design. When a company ventures into a domain in which it has no experience, it is undertaking an activity with high risk unless there are well-developed processes and expertise in place to enable it to make the transition.

☞ If such a company has weaknesses in areas such as estimating, tracking actuals to estimates, change control, and closing out action items, the risk would be higher.

☞ Even if no weaknesses were observed in these areas it could still be high risk if the company did not have a proven track record with the CMM KPAs or subprocess areas of process focus and training.

Project Size

If the project size for the new acquisition is significantly different from that of projects that the organization is used to working on, the organization may have problems adjusting their processes for the different scale. For example:

☞ Going from projects that can be developed with a single team to a project that requires coordinated efforts of several development teams may require new organization structures and reporting procedures.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 5   Create Critical Subprocess Area List
Guidelines   Importance of Individual Project Attributes in Determining Risk

☞ Intergroup communication may be a problem when the number or size of the groups increases if the organization does not have an established communications process.

☞ Existing facilities, tools, or procedures may not be able to handle a significantly larger volume of code.

☞ Established tracking and corrective action processes may not be adequate for a project with a shorter development schedule.

Type of Work

An organization that is effective at doing full development may not have processes that support tracking and coordinating work among multiple organizations.

An organization that has been doing only code development and not full development may not have processes that support all aspects of the development process.

Subcontractors

Subcontract management is not important on a development effort that includes no subcontractors. On the other hand this could be the most important KPA where critical parts of the software are expected to be subcontracted and the prime contractor will be the integrator.

The risk is compounded when multiple subcontractors are involved which is often the case with some of the very large DoD developments. In this case configuration management and requirements management (prime to subcontractor) become more significant.

Integration with the subcontractor's technical products may not be the only kind of integration involved. The subcontract schedules must be integrated into an overall project master schedule requiring project planning skills on the part of the prime.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 5   Create Critical Subprocess Area List
Guidelines   Importance of Individual Project Attributes in Determining Risk

Operational Precedence   The more unprecedented a system is, the more changes
                          there will be to the requirements. Therefore, it behooves
                          the acquisition agent to recognize the risk associated with
                          a contractor who has a poor track record in requirements
                          management, project management, configuration
                          management or project planning. These KPAs are
                          particularly important to maintaining control of cost and
                          schedule in the face of changing requirements.
                          Requirements management has less significance in an
                          environment where the requirements are well defined and
                          stable.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 5   Create Critical Subprocess Area List
Instructions How to Prepare a Key Issue Table

Instructions   **How to Prepare a Key Issue Table**

The SCE team prepares one Key Issue Table that applies to all development organizations in an evaluation. The form may be multiple pages depending on the number of development organizations and on the number of critical subprocess areas selected.

To prepare a Key Issue Table:

1. Make copies of a blank Key Issue Table form from Appendix E.

2. Enter the name of each development organization at the top of a column under the heading **Offerors**.

3. In the first column, enter the names of the critical subprocess areas and the corresponding KPAs as follows:

   a. Enter the name of a KPA that is part of the Target Process Capability.

   b. Under the KPA list the names of all of the subprocess areas of that KPA that were selected as critical subprocess areas.

      The subprocess areas should be indented to distinguish them from KPA names. Enter each one on a new line.

   c. Repeat the process until all of the KPAs in the Target Process Capability and all of the critical subprocess areas have been added. Multiple sheets may be required.

4. For each row with a critical subprocess area in column 1, enter the following information in the **Offerors** columns:

   a. For each subprocess area that was selected based on an attribute that was listed in the **Result** column of the Experience Table, enter the abbreviation for the attribute in that row under the column for each development organization that also had that attribute listed in the Experience Table.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 5   Create Critical Subprocess Area List
Instructions   How to Prepare a Key Issue Table

b. For each subprocess area that was selected because it is part of a nucleus software development capability, enter an asterisk in that row for every column under the **Offerors** heading.

c. For each subprocess area that was selected based on a lack of operational precedence, enter "Op" in that row for every column under the **Offerors** heading.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 5   Create Critical Subprocess Area List
Instructions How to Prepare a Key Issue Table

1 Make a copy of a blank Key Issue Table form from Appendix F.

2 Enter the names of the organizations to be evaluated in this row of headings

3 In this column, enter the names of the KPAs followed by the names of the critical subprocess areas. Indent the subprocess area names to distinguish them from the KPA names.

Key Issue Table

Critical Subprocess Areas	Offerors		
	Sigma Tech	Beverly Ind	Crystal City
Requirements Management			
Establish and maintain requirements baseline	Ps	Ps	Ps
Manage requirements-driven changes	Ps, *	Pt, Ps, *	Ps, *
Software Project Planning			
Develop estimates	Ps	Pt, Ps	Ps
Plan software activities	Ps	Ps	Ps
Make commitments	Ps, *	Ps, *	Ps, *
Software Project Tracking and Oversight			
Manage commitment changes			
Track progress	Ps, *	Pt, Ps, *	Ps, *
Take corrective action	Ps, *	Ps, *	Ps, *

4 In these columns, enter a symbol to indicate the significance of each subprocess area with respect to each organization. An abbreviation for an attribute indicates that the development organization has a lack of experience in that area and that the subprocess area is considered important to manage risk. An asterisk (*) indicates that the subprocess area is part of a nucleus software development capability. The abbreviation "Op" indicates that the subprocess area is important for managing risk due to a lack of operational precedence.

Figure 5-4: Preparing a Key Issue Table

Step 6   Originate Validation Worksheets

Objective

Record the set of critical subprocess areas for all development organizations on forms that can be used in subsequent information collection efforts.

Description

The Validation Worksheets are used by the SCE team to direct the investigations and to record the team consensus about each investigation topic. There is one worksheet for each critical subprocess area.

In this step, the SCE team creates a master set of worksheets that will be used to produce a set of worksheets for each specific site visit. One worksheet is created for each critical subprocess area selected.

The topics in this document most applicable to this step are listed below.

Chapter 5	Instructions, How to Prepare the Validation Worksheet	page 5-27

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 6   Originate Validation Worksheets
Instructions How to Prepare the Validation Worksheet

Instructions   **How to Prepare the Validation Worksheet**

A set of Validation Worksheets is generated for each site visit in the evaluation. There is at least one worksheet for each critical subprocess area. More pages may be required during the Specific Preparation Phase when the team enters the topic areas to be investigated for the subprocess area.

To create the Validation Worksheets:

1. Make copies of the blank Validation Worksheet form from Appendix E. You will need one form for each critical subprocess area.

2. For each form:

   a. Select a subprocess area from the critical subprocess areas listed on the Key Issue Table.

   b. Enter the name of the KPA in the box at the top of the first column.

   c. Enter the name of the subprocess area below the box.

3. Make a set of worksheets for each site.

Chapter 5   Preparing for the Site Visit
Section 1   General Preparation (Phase 2)
Step 6   Originate Validation Worksheets
Instructions How to Prepare the Validation Worksheet

1 Make copies of the blank Validation Worksheet form from Appendix F. Make one blank form for each critical subprocess area.

2 For each blank form, select a subprocess area from the list of critical subprocess areas listed on the Key Issue Worksheet.
Enter the names of the name of the KPA in this box.
Enter the name of the subprocess area below the box.

3 Make a copy of the set of forms for each organization to be evaluated.

Carnegie Mellon University
Software Engineering Institute

SCE Validation Worksheet

Projects: A._____   B._____   C._____   D._____

Software Project Planning

Develop Estimates

	Project	Explore Interview	Doc Review	Consolid Interview	Organization
	A				
	B				
	C				
	D				
	A				
	B				
	C				
	D				
	A				
	B				
	C				

Figure 5-5: Preparing a Validation Worksheet

Section 2   Specific Preparation (Phase 3)

Objective

Complete the portion of the site visit preparations for a specific development organization.

Participants

SCE Team Members

Time

2-3 days

Entry Criteria

❏ The steps of Phase 1 have been completed. The following outputs of Phase 1 are used in this phase:	page 4-14
❏ Target Product Profile	page 4-15
❏ Target Process Capability	page 4-16
❏ The steps of Phase 2 have been completed. The following outputs of Phase 2 are used in this phase:	page 5-2
❏ Mismatch Identification Tables	page 5-9
❏ Key Issue Table	page 5-23
❏ Validation Worksheets	page 5-27
❏ The SCE team has received the information requested from the development organizations. The following information is used in this phase:	page 4-7
❏ Proposed Project Profiles	page 4-8
❏ Project Profiles	page 4-9
❏ Organization Charts	page 4-10
❏ Responses to the Maturity Questionnaire (if used)	page 4-11

Activities

❏ Select Projects to Investigate (Step 7)	page 5-31
❏ Develop Key Issue Worksheet (Step 8)	page 5-34
❏ Develop Topic Lists (Step 9)	page 5-39
❏ Add Topics to Validation Worksheet (Step 10)	page 5-41
❏ Prepare for Exploratory Interviews (Step 11)	page 5-44
❏ Prepare Entry Briefing (Step 12)	page 5-52

Step 7   Select Projects to Investigate

Objective

Select the projects for evaluation that give the most insight into the processes that will be used.

Description

The SCE team is interested in identifying risks pertinent to the processes that will be used on the planned development. By evaluating the actual processes used on similar projects, the team obtains a clearer picture of the processes that will probably be used on the planned development.

The topics in this document most applicable to this step are listed below.

Chapter 5	Guidelines, Selecting Projects to Be Evaluated	page 5-32

Guidelines   **Selecting Projects to Be Evaluated**

The goal is to select those projects for on-site evaluation that will provide the most information about the development organization's software development capability relative to the proposed product development.

An initial selection of projects may be made by adding the 1's for the major attributes in each column of the Mismatch Identification Table. The projects with the highest sums are candidates for selection. The list of projects selected this way should be reviewed to make sure they are reasonable candidates. The following guidelines are provided to help in selecting projects:

Project Attributes   ☞ Select projects as close as possible to the application domain and product type of the product to be procured.

☞ Select some projects that are close in size to the product being procured even if they may not be in the same application domain.

☞ If the development organization proposes to use subcontractors on the development then select some projects where subcontractors are used.

☞ Try to select a set of projects that includes matches for all of the major attributes and as many of the minor attributes as possible.

Schedule   ☞ Select projects that are far enough along that they will provide data on all phases of the process that need to be evaluated. For example, a project that is only in the design phase will not provide information about how well coding and testing processes are being followed.

☞ Select projects that are current enough that the processes used will be applicable to the planned development project. Older projects are more likely to use processes that are not representative of those likely to be used on future projects.

☞ Avoid projects that have been completed or that will be completed by the time of the site visit. Once the teams have been disbanded, it may be impossible to get access to the team members for interviews.

Organization

☞ Select projects that have organizations similar to the organization proposed for the planned development.

☞ Select projects from the same organization unit (division, department, etc.) as the one proposed for the planned development.

Location

☞ If possible, select projects that have teams normally located at the site where the evaluation will be conducted. If the development organization has to bring people from remote locations it is usually not possible to select random team members to interview.

Access

☞ Try to avoid projects where the team may not be granted ready access to the project information such as defense contracts that are highly classified or projects that are company confidential.

Step 8   Develop Key Issue Worksheet

Objective

Create a consolidated list of key issues for investigation at the development organization site.

Description

The SCE Team needs to determine how to prioritize the subprocess areas that are to be investigated at each site. Although the same list of critical subprocess areas is investigated for all development organizations, each organization has unique strengths and weaknesses. More time or effort should be spent investigating the critical subprocess areas that correspond to areas where the team has reason to suspect an area of weakness for a particular development organization.

Indications of weakness may come from:

- A lack of experience with some aspect of the project indicated by the Experience Tables.

- From inconsistencies and anomalies in the responses to the Maturity Questionnaire.

In this step, the SCE team determines the key issues that apply to a specific development organization. This is done by:

1. Preparing the Questionnaire Worksheet.

2. Preparing the Key Issue Worksheet.

The topics in this document most applicable to this step are listed below.

Chapter 5	Instructions, How to Prepare the Questionnaire Worksheet	page 5-35
	Instructions, How to Prepare the Key Issue Worksheet	page 5-36

Chapter 5   Preparing for the Site Visit
Section 2   Specific Preparation (Phase 3)
Step 8   Develop Key Issue Worksheet
Instructions How to Prepare the Questionnaire Worksheet

Instructions   **How to Prepare the Questionnaire Worksheet**

[Add when the new Questionnaire Worksheet is available]

Chapter 5   Preparing for the Site Visit
Section 2   Specific Preparation (Phase 3)
Step 8   Develop Key Issue Worksheet
Instructions How to Prepare the Key Issue Worksheet

Instructions   **How to Prepare the Key Issue Worksheet**

A Key Issue Worksheet is prepared for each specific development organization. The form may be multiple pages depending on the number of projects and on the number of critical subprocess areas.

To prepare a Key Issue Worksheet:

1. Make copies of the blank Key Issue Worksheet form in Appendix E.

2. In the first column, copy the names of the KPAs and subprocess areas from the first column in the Key Issue Table.

3. Enter the name of the development organization in the heading box at the top of the second column. Copy the information from the column of the Key Issue Table that contains the entries for the same development organization.

4. Enter the name of each project to be evaluated at the top of a column under the heading **Projects**.

5. In the columns for each project, record the results of the evaluation of the Questionnaire Worksheet responses for the specific development organization. The questions on the Questionnaire Worksheets are grouped into subprocess areas, so the results can be directly entered into the appropriate row (critical subprocess area).

   a. Analyze the responses for inconsistencies and anomalies. Both inconsistencies and anomalies raise issues that the team may want to investigate.

      Inconsistencies refer to contradictory responses for two different questions for the same project.

      Anomalies refer to responses made to the same question that are different for the selected projects.

   b. Record an inconsistency by entering the abbreviation "Inc" followed by the identification numbers for the questions with inconsistent

Chapter 5   Preparing for the Site Visit
Section 2   Specific Preparation (Phase 3)
Step 8   Develop Key Issue Worksheet
Instructions How to Prepare the Key Issue Worksheet

responses in the row for the corresponding subprocess area under the column for the project that had the inconsistency.

c. Record an anomaly by entering the abbreviation "An" in each column of the row for the corresponding subprocess area. Enter the response that was given by each project into the same row under the column corresponding to that project. Enter the question's identification number in the same row under the first column.

Chapter 5   Preparing for the Site Visit
Section 2   Specific Preparation (Phase 3)
Step 8   Develop Key Issue Worksheet
Instructions How to Prepare the Key Issue Worksheet

1 Make a copy of a blank Key Issue Worksheet form from Appendix F.

2 Copy the first column of the Key Issue Table to this column.

3 Enter the name of the development organization in the heading box at the top of the second column.

Copy the column from the Key Issue Table that contains the entries for the same development organization.

4 Enter the names of the projects to be evaluated in this row of headings

Key Issue Worksheet

Critical Subprocess Areas	Sigma Tech	Projects		
		Able	Baker	Charlie
Requirements Management				
Establish and maintain requirements baseline	Ps			
Manage requirements-driven changes	Ps, *			
Software Project Planning				
Develop estimates	Ps			Inc: est. training
Plan software activities	Ps			
Make commitments	Ps, *			
Software Project Tracking and Oversight				
Manage commitment changes		Customer I/F	Customer I/F	
Track progress	Ps, *			
Take corrective action	Ps, *	issue trking		issue trking

5 In these columns, record the results of the evaluation of the Questionnaire Worksheet responses for each of the projects.

Figure 5-6: Preparing a Key Issue Table

Step 9   Develop Topic Lists

Objective

Select topics for probing the process implementation; topics define the observable work practices that map to the critical subprocess areas.

Description

Subprocess areas represent too large an area to be the subject matter for one interview question or for the review of one document. Therefore, the SCE team must develop a probing strategy for investigating each subprocess area.

Topics provide the subject matter focus for the document review and interviews for a single subprocess area. They enable the SCE team to focus attention on the parts of each subprocess area that are of particular interest to the team for this specific development organization. Topics serve as a guide to the team to help them develop questions that are open-ended and specifically focused.

The topics are created using the features listed in Appendix B. The features are derived from the common features defined in the CMM. They represent characteristics that apply to any process. When a feature is paired with a subprocess area it becomes a topic.

In this step, the SCE team develops the list of topics as follows:

1. Each team member develops an individual list of topics

2. The team creates a consolidated list of topics from the individual topic lists.

The topics in this document most applicable to this step are listed below.

| Chapter 5 | Instructions, How to Select Topics | page 5-40 |

Instructions   **How to Select Topics**

Topics are selected from the list of features listed in
Appendix B.

1. Each team member creates a list of topics as follows:

    a. For each row in the Key Issue Worksheet (critical
        subprocess area) evaluate the key issues.

    b. Select the features from the list in Appendix B
        that probe the key issues for that subprocess
        area. If there are no key issues, select features
        that will provide a means of starting the
        investigation for the subprocess area.

    c. Record the feature selections on an informal list.

2. Consolidate the individual topic lists into a single list as
    follows:

    a. Merge the individual topic lists into a single list.

    b. As a team, rank the topics for each subprocess
        area in order of importance.

    c. Draw a cut line to prioritize time to best achieve
        site visit goals.

Step 10   Add Topics to Validation Worksheet

Objective   Capture the consolidated topic list for use at a particular
                site.

Description   During the site visit, the SCE team members will use the
                Validation Worksheets to document their observations
                about each topic. The worksheets provide a convenient way
                to consolidate information about the selected topics for
                each subprocess area.

                In Step 6 (<span>page 5-26), a set of Validation Worksheets was
                created for each site to be visited. Each worksheet
                contained the name of a critical subprocess area and the
                associated KPA. In this step, the SCE team adds projects to
                be investigated during the site visit and the topics from the
                consolidated list of topics for the corresponding subprocess
                area to each worksheet.

                The topics in this document most applicable to this step are
                listed below.

Chapter 5	Instructions, Adding Topics to the Validation Worksheet	page 5-42

Chapter 5   Preparing for the Site Visit
Section 2   Specific Preparation (Phase 3)
Step 10   Add Topics to Validation Worksheet
Instructions   Adding Topics to the Validation Worksheet

Instructions   **Adding Topics to the Validation Worksheet**

Complete the preparation of the Validation Worksheets
created in Step 6 (➥page 5-26) by doing the following:

1. Select a set of Validation Worksheets created in Step 6.

2. Enter the names of the projects to be evaluated at the
   top of each page of the set of worksheets for the site.

3. For each critical subprocess area (each worksheet page),
   enter the names of the topics selected for the site in the
   rows of the first column.

Chapter 5   Preparing for the Site Visit
Section 2   Specific Preparation (Phase 3)
Step 10   Add Topics to Validation Worksheet
Instructions Adding Topics to the Validation Worksheet

1 Select a set of the Validation Worksheets created in Step 6.

2 Enter the names of the projects to be investigated in this row.

3 Enter the topics to be investigated for this subprocess area in the first column.

Carnegie Mellon University
Software Engineering Institute

SCE Validation Worksheet

Projects: A. _Able_   B. _Baker_   C. _Charlie_   D. _____

Software Project Planning	Project	Explore Interview	Doc Review	Consolid Interview	Organization
Develop Estimates					
Comments: look at both cost and size estimates	A				
	B				
	C				
	D				
organizational policies	A				
	B				
	C				
	D				
organizational structures	A				
	B				
	C				

Figure 5-7: Preparing a Validation Worksheet

Step 11   Prepare for Exploratory Interviews

Objective

Develop a detailed interview strategy including the team's decision on who will be interviewed, when they will be interviewed, and what they will be asked.

Description

In this step, the team develops a high-level interview strategy and prepares materials to guide them during the interviews. The activities of this step include:

1. Preparing the interview plan.

2. Creating Interview Worksheets.

The interview plan does not have a particular format. It includes the following information:

- The list of potential interview candidates specified by position in the organization.

- The order in which the candidates are to be interviewed.

- How much time is allocated to each interview.

- The topics to be investigated for each interview.

The Interview Worksheets contain the specific lead-in questions to be asked. The worksheets help focus the team on the relevant issues during the interview, increasing the chances of gathering the relevant information during the interview.

General guidance for planning and conducting interviews is provided in Chapter 7. Specific guidance which is applicable to this step is provided in this section. The topics in this document most applicable to this step are listed below.

Guidelines **Preparing the Interview Plan**

There are two strategies that teams have found to be
effective for establishing the interview plan:

- Working "top down" through the organization.

- Interviewing project by project.

Sample site visit schedules that demonstrate both of these
strategies are shown in Figure 5-8 (➡page 5-47) and Figure
5-9 (➡page 5-48).

The schedules demonstrate the interrelationships between
the document review, interviewing, and caucusing
activities during the Site Data Collection phase. For both of
the example schedules, it is assumed that the formal
Findings Report (Step 23) is prepared later, after the site
visit is completed.

The first schedule assumes that interviews are structured
"top down," interviewing all of the project managers, then
the software supervisors, etc. The second assumes that the
projects are interviewed sequentially—first Project A, then
Project B, and so on.

The sample schedules show general blocks of time allocated
for interviews. These blocks need to be planned out in
detail showing who is to be interviewed and how much time
is allowed for each interview. Time should be allowed after
each interview for a brief caucus to discuss what was
learned and to review the plan for the next interview.

The original strategy may be altered during the site visit
depending on what the team finds during the initial
document review. If, for example, organizational policies,
procedures, roles, and responsibilities are easy to identify
from documentation, the team may require less interview
time and may prefer to spend time gathering "audit trail"
information. On the other hand, if documentation is

Day	Activity	Steps	Hours
Day 1	Initial organization meeting with site management SCE team in-brief (15 minutes) Development organization in-brief/ Selected project presentations (60 minutes) SCE team caucus (15 minutes)	13	1.5
	Initial document review and caucus on documentation. (Documents should be available in assigned meeting room.)	14, 16	3.0
	Exploratory interviews with project managers and software managers, with caucuses between each.	15, 16	3.0
Evening	Document Review and Caucus	17, 16	3.0
Day 2	Exploratory interviews continue with software supervisors, SQA engineers, SCM personnel, test personnel, and software engineers	15, 16	3.5
	Review of documents requested during exploratory interviews	17	2.0
	Caucus on information gained, possibly with interviews of people who create track record-level documentation.	16, 15	2.0
Evening	Preparation of Preliminary Findings	18	1.5
	Development of Consolidation Plan	19	1.5
Day 3	Consolidation interviews	20	2.0
	Final Document Review	21	2.5
	Determination of Findings	22	1.5
	Exit Briefing	24	2.0

Figure 5-8: Site Visit Schedule, Example 1.
Interviews Conducted "Top - Down"

Day	Activity	Steps	Hours
Day 1	Initial organization meeting with site management SCE team in-brief (30 minutes) Development organization in-brief/ Selected project presentations (60 minutes)	13	1.5
	Initial document review, caucus on documents	14, 16	2.0
	Exploratory interviews with Project A, caucus	15, 16	1.5
	Exploratory interviews with Project B, caucus	15, 16	1.5
	Exploratory interviews with Project C, caucus	15, 16	1.5
Evening	Document review and caucus	17, 16	3.0
Day 2	Document review and caucus	17,16	1.0
	Exploratory interviews with Project D, caucus	15, 16	1.5
	Site SQA interview and caucus	15,16	0.75
	Site Software CM interview and caucus	15,16	0.75
	Corporate management interview and caucus	15,16	0.75
	Site SEPG interview and caucus	15,16	0.75
	Document review and caucus	17,16	2.5
Evening	Preparation of Preliminary Findings	18	1.5
	Development of Consolidation Plan	19	1.5
Day 3	Consolidation interviews	20	2.0
	Final Document Review and caucus	21,16	1.5
	Determination of Findings	22	1.5
	Exit Briefing	24	2.0

Figure 5-9: Site Visit Schedule, Example 2.
Interviews Conducted One Project at a Time

complex or unorganized, more interview time may be needed to clarify the organization's processes. The team must be able to react to unforeseen circumstances.

Coordinating interview schedule.

Once the interview schedule has been developed it should be coordinated with the development organization. This is normally done through the development organization's point of contact. Refer to "Logistic Arrangements" (➥page 6-6) for more information.

To maintain fairness, each development organization must be given the same amount of time to prepare for their site visit. Therefore, the exact schedule for a development organization will not be determined until that development organization has been notified of the site visit dates. Refer to "Scheduling a Site Visit" (➥page 6-3) for more information.

The purpose of coordinating the interview schedule with the development organization is to make sure that the organization can accommodate the plan. Some of the people that the team wants to interview may have to be brought in from other locations or may have other commitments to work around. Coordinating the schedule with the organization can help the team to achieve their objectives while having the minimum impact on the organization.

Instructions

How to Prepare Interview Worksheets

The SCE team creates an Interview Worksheet for each person they plan to interview. More than one sheet may be required for each interview candidate. To create an Interview Worksheet:

1. Make a copy of the blank Interview Worksheet form in Appendix E.

2. Identify the interview candidate by entering the person's position in the space provided at the top of the form. Include the project name if applicable. The other information in the top of the form will be completed at the time of the interview.

3. Record the planned interview questions in the first column as follows:

   a. For each interview topic, enter the name of the KPA followed by the name of the subprocess area. Indent the subprocess area name to distinguish it from the KPA name. This information comes from the Validation Worksheets and is used to relate the interview results back to the Validation Worksheets.

   b. Under the KPA and subprocess area name list the questions planned for the topic.

   c. Under the questions, record any additional notes that you want to refer to during the site visit such as possible document types to request.

Figure 5-10: Preparing a Validation Worksheet

Step 12   Prepare Entry Briefing

Objective

Establish the agenda and briefing for the Initial Organization Meeting and set initial expectations for the site visit.

Description

The first activity of the SCE team during a site visit is to give an entry briefing to the organization being evaluated. This is done during the Initial Organization Meeting. The briefing is prepared as part of the site specific preparations.

The purpose of the entry briefing is to introduce the team members to the organization and to set the expectations for the rest of the site visit. The entry briefing should be no more than 30 minutes.

Topics for the briefing may include:

- A brief introduction of the team members, and an appropriate description of team qualifications to conduct this SCE.

- What the team hopes to accomplish during the visit.

- A description of the major on-site activities.

- The schedule for the site visit activities.

- The ground rules the team intends to follow.

- The KPAs that the team will investigate during the evaluation (the Target Process Capability).

- A sample of how findings will be reported.

- What information, if any, the team will present at the end of the site visit.

The team briefing should be standardized for all development organizations.

The Introductory meeting should not be used to educate the developer about SCE. The developer should have been briefed on SCE prior to the site visit.

Chapter 6   Conducting the Site Visit

Chapter 6   Conducting the Site Visit

This chapter provides detailed instructions for the activities that SCE team members perform during a site visit. It covers Step 12 through Step 24 of the SCE method. These steps make up the Site Data Collection Phase (Phase 4) and the Findings Phase (Phase 5) of the method.

Section 1   Arranging the Site Visit

Part of preparing to conduct an SCE includes coordinating with the development organization being evaluated to make the necessary arrangements for the site visit. A lack of proper coordination ahead of time could impact the effectiveness of the evaluation.

SCE teams have a lot to accomplish during a site visit. There is no time to complete arrangements that should have been taken care of ahead of time.

The development organizations usually realize the importance of a successful site visit. They will generally try to accommodate the needs of the SCE team if they know what the needs are.

The following checklist shows some of the key activities of arranging a site visit.

Checklist for
Coordinating Site Visit
Arrangements

At least 2 months prior to site visit	
❑ Negotiate date for site visit	page 6-3
❑ Ask the development organization to identify a point of contact	page 6-5
❑ Notify development organization of logistic requirements	page 6-6
At least 2 weeks prior to the site visit	
❑ Let the development organization know which projects have been selected for evaluation.	
❑ Request documents for initial document review	page 6-7
❑ Coordinate the initial interview plan	page 6-8
❑ Coordinate agenda for initial meeting	page 6-9
❑ Ask the development organization to identify a site visit coordinator	page 6-6

Scheduling a Site Visit

The site visit should be scheduled far enough in advance so that both the SCE team and the development organization have sufficient time to prepare. When SCE is used as part of the source selection process, fairness to all development organizations must be considered. The following guidelines are provided for scheduling the site visit:

☞ The development organization should know at least 60 days ahead of time when the site visit will occur.

The development organization will need to make key personnel available for the interviews. Generally, the SCE will be looking at multiple projects so the impact

will be wide spread. They will need advance notice so that these people can work their other commitments around the SCE schedule.

They will also need some time to make space available for the SCE team to conduct the evaluation.

☞ The preparation schedule should contain slack time to allow for reasonable delays so that the site visit will not need to be rescheduled.

Particularly when SCE is used in a source selection, it is important to stick to the established schedule so that each development organization has the same time to prepare.

☞ A site visit is normally scheduled for three days.

The site visit may be conducted in a normal work week with a day before and after the visit for travel time.

☞ Back to back site visits are not recommended.

An SCE site visit can be very stressful for the team members. When SCEs are to be conducted on more than one site, teams should have at least one week between site visits.

☞ When an SCE is conducted as part of a source selection process, each development organization must be given the same amount of time to prepare for their site visit.

Generally, the development organizations are informed at the same time, by means of the acquisition announcement and the request for proposals, that an SCE will be conducted. Therefore, site visits for all offerors should be scheduled relatively close in time so that no organization has more time to prepare.

Arrangements for each individual site visit should be made according to the same schedule so that each site has the same amount of time to prepare. In particular, make sure that each site gets the same amount of notice regarding:

• Which projects have been selected.

• What types of documents are needed for initial document review.

- What positions are included in the initial interview plan.

- What information to present at the initial organization meeting.

☞ When using SCE in a source selection, the responses to the Maturity Questionnaire are generally requested as part of the proposal (➡page 4-7). If the questionnaire is sent out later, the following issues should be considered:

- The date for the site visit should be negotiated before shipping the questionnaire.

- Each offeror must have the same number of working days between being asked for responses to the Maturity Questionnaire and the site visit.

- The developer should have a minimum of one week to respond to the Maturity Questionnaire.

- The team will need at least two working days to carry out their final preparations for the site visit after receiving the questionnaire responses.

- Factor in time for shipping of the questionnaire and responses.

Point of Contact

The SCE team leader should ask the development organization to identify a point of contact for the team to work with to make arrangements for the site visit. The SCE team should work with the designated point of contact regarding the following:

- Logistic arrangements for the site visit (➡page 6-6)

- Request of documents for initial document review (➡page 6-7)

- Coordination of the interview schedule (➡page 6-8)

- Content of the development organization's entry briefing (➡page 6-9)

Site Visit Coordinator   The SCE team leader should ask the development
organization to identify a site visit coordinator that the
team will work with during the site visit. The site visit
coordinator may or may not be the same person as the point
of contact identified prior to the site visit. If it is not the
same person, the final site visit arrangements should be
reviewed with the site visit coordinator prior to the site
visit.

The SCE team will work through the site visit coordinator
to:

• Coordinate changes to the interview schedule.

• Request additional documents to review.

• Arrange administrative details.

Logistic Arrangements   Logistic arrangements for the site visit should be
coordinated with development organization's point of
contact:

☞ A meeting room is needed that is large enough to
accommodate at least ten people comfortably.

During the interviews there will only be the 4 to 6 team
members and one interview candidate in the room.
However, there may be occasions where the team will
want to meet with several people at once to discuss
planning and coordination of the site visit.

☞ If the team has a preferred seating arrangement for the
interview, make sure that the room can be arranged to
accommodate it.

☞ A work area is needed for team caucuses and document
review.

Some teams prefer to use separate rooms for
interviewing and document review. Other teams use
one large room for both.

Tables with an adequate work surface are needed for the document reviews. The room should contain bookcases or filing cabinets to help the team manage the large number of documents they will be reviewing.

☞ The team must have exclusive use of the rooms for the duration of the site visit.

☞ The team will need access to telephones and copy facilities during the site visit.

☞ Arrangements must be made for access to the facilities.

If team members are required to check in each morning, then the schedule must allow for it. Team members will generally need to work later than normal working hours.

Requesting Documents for the Initial Document Review

One of the first activities the SCE team will perform during the site visit is the initial document review. The documents to be reviewed during this time should be requested prior to the site visit so that they will be available when the team needs them.

The following are guidelines for requesting documents for the initial document review:

☞ Provide the developer's point of contact with a list of the documentation desired for the initial document review. Request the following types of documents:

  • Updates to organization charts.

  • Copies of the organization's policies and procedures applicable to their software process capability.

  • Copies of the specific procedures as tailored for the selected projects.

  • Detailed process standards/directives for each of the selected projects.

☞ Request documents by type or description, not by title.

☞ Ask the developer to have the documentation in the meeting room at the start of the site visit.

☞ Request the documents at least two weeks prior to the site visit to give adequate time to respond.

☞ Do not request the documents to be sent for review prior to the site visit because there will generally not be time to review them and the volume will be too large.

☞ Request ahead of time only documentation the SCE team has time scheduled to review and factor into its planning activities.

☞ When SCE is used in a source selection, give each development organization the same description of the documents needed and allow the same amount of time to collect the documents.

The amount of documentation which an organization usually makes available for the initial document review can be overwhelming. The team is not expected to look at every page of each document. The team will look for specific information applicable to the critical subprocess areas.

Coordinate Initial Interview Plan

The initial interview plan developed in Step 11 (➡page 5-44) should be discussed with the development organization's point of contact at least 2 weeks before the scheduled site visit (➡page 6-8). The purpose of the discussion is to verify that the people the team plans to interview will be available and to give the development organization a better idea of what affect the site visit will have on their ongoing work.

Some of the people that the team wants to interview may have to be brought in from other locations or may have other commitments to work around. Coordinating the schedule with the organization can help the team to achieve their objectives while having the minimum impact on the organization.

The team leader should make it clear that the initial interview schedule is subject to change as the team gathers information during the site visit. The team does not want to be limited to the interview candidates identified in the initial plan.

Coordinate Agenda for Initial Organization Meeting

The SCE team leader should discuss the agenda for the initial organization meeting with the development organization's point of contact. The development organization will want to understand the purpose of the meeting so that they know who should attend.

The SCE team may ask the development organization to present a brief summary of their organization and their software development process during the initial organization meeting. Such a summary can help to acquaint the team with the development organization's terminology and to orient them to the structure of organization's process documents.

The SCE team leader should provide the development organization's point of contact (➡page 6-5) with a general outline that shows the information the team wants to hear in the entry briefing. The team does not want a marketing pitch. The team does not want to be subjected to information that will not help them in their evaluation.

The time for the organization's presentation should be explicitly limited to approximately 60 minutes. The total time for the initial organization meeting should be no more than 90 minutes.

The development organization should explain to the SCE team

- What the organization does (without giving a "marketing pitch" or an in-depth recital of their standard processes).

- The organization structure, (who does what), especially any changes that have occurred since the initial organization charts and information that was provided.

- How responsibility, accountability, and authority are managed particularly in regard to such items as software configuration management, software quality assurance, integration and test, requirements definition, systems test, and software development.

- How the organization's process integrates responsibility, accountability, and authority through the development lifecycle; the organization's description should be focused on the projects selected for review.

- The relationship the proposed project has to the rest of the organization in terms of procedures to be followed, reporting relationships, etc.

- There is no place in the initial organization meeting for discussion. Its purpose is solely information exchange pertinent to the site visit.

Section 2 — Site Data Collection (Phase 4)

Objective

Collect the objective evidence on which to base the findings.

Participant

SCE Team Members

Time

3 days

Entry Criteria

❑ The steps of Phase 3 have been completed. The following outputs of Phase 3 are used in this phase:	page 5-29
❑ Validation Worksheets	page 5-41
❑ Interview Worksheets	page 5-44
❑ Interview Schedule	page 5-44
❑ Entry Briefing	page 5-52
❑ The site visit has been arranged	
❑ The development organization has designated a site visit coordinator	page 6-6
❑ The site visit has been scheduled	page 6-3
❑ Logistic arrangements have been coordinated	page 6-6
❑ Documents for the initial document review have been requested	page 6-7
❑ The initial interview schedule has been coordinated	page 6-8
❑ Agenda for the Initial Organization Meeting has been coordinated	page 6-9

Activities

❏ Conduct Initial Organization Meeting (Step 13)	page 6-13
❏ Conduct Initial Document Review (Step 14)	page 6-15
❏ Conduct Exploratory Interviews (Step 15)	page 6-18
❏ Hold Team Caucus (Step 16)	page 6-20
❏ Conduct Document Review (Step 17)	page 6-24
❏ Develop Preliminary Findings (Step 18)	page 6-24
❏ Create Consolidation Plan (Step 19)	page 6-27
❏ Conduct Consolidation Interviews (Step 20)	page 6-30
❏ Conduct Final Document Review (Step 21)	page 6-31

Step 13   Conduct Initial Organization Meeting

Objective   Clarify the expectations of the site visit.

Description   The initial organization meeting usually consists of:

1. an entry briefing by the SCE team, and

2. a presentation by the development organization.

The SCE team leader presents the entry briefing developed in Step 12 (➡page 5-52). The entry briefing introduces the team members and describes what the team expects to accomplish. The entry briefing should be no more than 15 to 30 minutes.

The development organization then makes its presentation to the SCE team. The presentation should follow the guidelines provided to the development organization's point of contact prior to the site visit (➡page 6-9). It should take no more than one hour.

The team leader should ensure that the organization's presentation follows the established agenda. It may be necessary to politely stop the presentation if it does not contain the requested information or if the presentation goes on too long.

The team should look for information in the development organization's presentation that they can factor into their site visit plans such as:

- Updates to organization charts.

- Clarification of roles.

- Documents to review.

- Interview candidates.

At this time, the team should confirm that the previously negotiated arrangements for facilities have been made correctly (e.g., working space, meeting rooms, telephone access), that requested documentation is available, and

that the right people are available for preliminary interviews. These items were requested from the development organization's point of contact during the Specific Preparation phase.

Step 14   Conduct Initial Document Review

Objective

Determine the degree to which the organization and project-level documentation define and support standard processes for the KPAs and subprocess areas under investigation.

Description

The initial document review helps the team gain a better understanding of the working environment the development organization provides from a process view point. It helps the team to refine the preparations already made for exploratory interviews. By providing further insight into the policies and procedures that guide the organization's processes, the team can sometimes eliminate the need for a question during the interviews or sharpen the focus for a question.

General guidance for conducting document reviews is provided in Chapter 8. Specific guidance which is applicable to this step is provided in this section. The topics most applicable to this step are listed below.

Chapter 6	Guidelines, What to Look For During Initial Document Review	page 6-16
Chapter 8	Section 1, Types of Documents	page 8-2
	Section 2, What to Review	page 8-4
	Section 3, What to Look For	page 8-6
	Guidelines, Reviewing Organization Level Policies and Directives	page 8-8
	Guidelines, Reviewing Project Standards and Procedures	page 8-10
	Guidelines, Reviewing Evidence of Process Activity	page 8-12
	Section 4, How to Record Results	page 8-14

Guidelines   **What to Look For During Initial Document Review**

During the initial document review the SCE team reviews the documents that were requested prior to the site visit. The team should be looking at:

- updates to organization charts

- organization level policies and procedures applicable to their software process capability

- plans, standards, and procedures for the selected projects

- detailed process standards/directives for the selected projects

The development organization may provide more than the team asks for. In that case the team will need to sort out the documents that are most important for the initial document review. Instead of spending a lot of time trying to find the documents, the team should ask the site coordinator to locate the documents for them. The team should ask for the documents by description rather than by name.

The initial document review should focus on the following objectives:

☞ Identify the organization level documents that define the organization's software development process

☞ Understand the relationship among the organization level documents and the project specific documents

☞ Understand the organizations process for defining the software development process on individual projects

☞ Clarify the roles and responsibilities of the people scheduled for interview and adjust the interview plan as needed

☞ Begin to identify the documents associated with each of the subprocess areas being investigated

During the initial document review it is important not to get bogged down in any one area. If it seems as if it is taking too long to locate or to understand a particular piece of information, it may be better to move on to something else. The exploratory interviews may help to identify where the information is or to clarify the information in the documents.

During the initial document review, the team may discover information that will help to refine the plans for the rest of the site visit. The information may answer some of the questions planned for exploratory interviews or may lead to other questions. The team may also gain a more accurate view of the roles and responsibilities within the organization and may discover that changes to the interview plan are needed.

Occasionally, some topics may be partly or fully validated through the initial document review. Validation requires team consensus that at least two pieces of evidence support the finding. When this happens the team may reallocate the time for investigating these topics to other areas.

In general though, the existence of a policy or procedure does not constitute a finding. Evidence must be found that the document is used. Exploratory interviews are needed to determine if the right people know the policy or procedure and to help the team to locate the trail of information that shows the process is used.

Step 15   Conduct Exploratory Interviews

Objective

Provide insight into how the subprocess areas are implemented in practice; determine the extent that processes have been internalized by the development organizations; identify critical implementation-level documents.

Description

SCE team typically spends the majority of time in a site visit conducting exploratory interviews with key personnel from the development organization. The interviews help the team to gain an understanding of the organization's software development processes. The primary goal is to discover the objective evidence that will show that the processes are being followed.

General guidance for conducting interviews is provided in Chapter 7. Specific guidance which is applicable to this step is provided in this section. The topics most applicable to this step are listed below.

Chapter 6	Guidelines, Conducting Exploratory Interviews	page 6-19
Chapter 7	Section 1, The Interview Environment	page 7-2
	Section 2, Team Member Roles	page 7-5
	Section 6, Controlling the Interview	page 7-17
	Section 3, Selecting Interview Candidates	page 7-8
	Section 5, Developing Interview Questions	page 7-15
	Section 7, How to Record Results	page 7-19

Guidelines   **Conducting Exploratory Interviews**

The exploratory interviews are conducted according to the interview plan developed in Step 11 (➡page 5-44). The plan contains the schedule of who will be interviewed specified by project and position.

There should be an Interview Worksheet (➡page 5-50) for each interview candidate. The Interview Worksheet should contain the planned questions for the interview candidate. Space should be provided to record the responses (➡page 7-19) and to record follow-up questions that may be asked.

Team members should be assigned specific roles (➡page 7-5) for exploratory interviews. The roles do not need to be the same for each interview but team members must clearly understand what their roles are for each interview session.

SCE teams should make every effort to stick to the interview schedule.

A brief team caucus (➡page 6-20) should be held after every interview to obtain team consensus about what the team heard. The caucus helps to keep the team focused on the objectives. It also gives the team a chance to make changes to the Interview Worksheet for the next interview candidate.

Step 16   Hold Team Caucus

Objective

Analyze, share, and consolidate information in order to reach conclusions about topics.

Description

The team caucus is a decision making activity that supports the data gathering activities of document review and exploratory interviews. The caucus provides a chance for team members to discuss the data that was gathered while it is still fresh in everyone's minds. It helps to keep the team in agreement on what has been learned and on what data needs to be collected. The caucus keeps the team focused on the objectives of the SCE.

General guidance for conducting an effective team caucus is provided in Chapter 9. Specific guidance which is applicable to this step is provided in this section. The topics most applicable to this step are listed below.

Chapter 6	Guidelines, Holding a Team Caucus	page 6-21
Chapter 9	Section 1, Team Member Roles	page 9-2
	Section 2, Consensus Process	page 9-5
	Section 3, Adjusting the Site Visit Plan	page 9-6
	Section 4, Assessing the Information Collected	page 9-8
	Section 5, Findings	page 9-13

Guidelines   **Holding a Team Caucus**

Brief team caucus sessions should be scheduled for after
each interview and after any document review sessions.
These caucus sessions should b̶ ̶b̶᷇᷇᷇t five or ten minutes
long. Longer caucus session᷇   ᷇c scheduled once or
twice a day to review the plans for the rest of the site visit
and to make any necessary changes.

During caucus, the team assesses their progress toward the
goal of validating topics by evaluating the information
gathered so far. No particular format is specified for the
caucus, but the following steps are typical:

- The team reviews information collected about the topics
  that were the focus of the most recent investigations.

- The team reviews any significant new information, and
  identifies areas that require further clarification.

- If the team consensus shows that the information is
  sufficient for a candidate finding, the candidate finding
  is appropriately defined and then entered on the
  Validation Worksheet for review during Step 18,
  "Develop Preliminary Findings" (➡page 6-24).

- If the team cannot reach consensus, they identify
  information that will settle the outstanding issues.

Step 17   Conduct Document Review

Objective   Search for objective evidence of how processes are
                   implemented at the working level.

Description   Document reviews are conducted following a set of
                   interviews to give the team a chance to validate what they
                   have heard.

                   General guidance for conducting document reviews is
                   provided in Chapter 8. Specific guidance which is
                   applicable to this step is provided in this section. The topics
                   most applicable to this step are listed below.

Chapter 6	Guidelines, Document Reviews	page 6-23
Chapter 8	Section 1, Types of Documents	page 8-2
	Section 2, What to Review	page 8-4
	Section 3, What to Look For	page 8-6
	Guidelines, Reviewing Organization Level Policies and Directives	page 8-8
	Guidelines, Reviewing Project Standards and Procedures	page 8-10
	Guidelines, Reviewing Evidence of Process Activity	page 8-12
	Section 4, How to Record Results	page 8-14

Guidelines   **Document Reviews**

Document reviews give the team a chance to validate the information that they hear in the interviews. The purpose of this step is to search for objective evidence of how the processes are implemented at the working level—this provides support for findings. In other words, the team determines whether the processes defined on paper and elicited from the interviews correspond to what the people on the projects are actually doing.

The team reviews project-level and implementation-level documents for a project to validate information gathered through other sources such as interviews and higher level document review. The topics on the Validation Worksheets and the results of the interviews as recorded on the Interview Worksheets are used to focus the review.

Informal document review working notes are kept to use during the caucus; the relevant information is entered onto the Validation Worksheet after caucusing in Step 16 ().

Documents on this level provide an audit trail of the processes used and the work performed. Through these reviews, the team confirms or negates the proposition that the actual work practices implement the processes described in the organization- and project-level documents.

This level of document review focuses on implementation-level documents; but some project- and organization-level documents may also be referenced.

Pairs of team members may visit the organization's document library, if one exists. Some team members may prefer to select the documents they review from the library of documents rather asking for specific documents. However, in any document review, the objective is to collect objective evidence about the critical subprocess areas by investigating the topics on the Validation Worksheets.

Step 18   Develop Preliminary Findings

Objective

Articulate conclusions about the subprocess areas based on the information available; guide subsequent information gathering efforts.

Description

Step 18 is a special purpose team caucus in which the SCE team assesses what information has been collected about each subprocess area being investigated. The team determines what conclusions can be reached about each of the topics investigated for each of the critical subprocess areas. If the team can not reach consensus on a topic or if the team does not have enough objective evidence, they determine what additional information is needed.

General guidance for conducting an effective team caucus is provided in Chapter 9. Specific guidance which is applicable to this step is provided in this section. The topics most applicable to this step are listed below.

Chapter 6	Instructions, Developing Preliminary Findings	page 6-25
Chapter 9	Section 1, Team Member Roles	page 9-2
	Section 2, Consensus Process	page 9-5
	Section 3, Adjusting the Site Visit Plan	page 9-6
	Section 4, Assessing the Information Collected	page 9-8
	Section 5, Findings	page 9-13

Instructions   **Developing Preliminary Findings**

The Validation Worksheets contain the topics for investigation. These worksheets were developed in Step 6 (➥page 5-26) and Step 10 (➥page 5-41). There is one worksheet for each subprocess area being investigated. Each worksheet contains the topics being investigated for the subprocess area.

Review the Validation Worksheets one at a time. For each investigation topic on the worksheet, consider the applicable information collected during the exploratory interviews and document reviews.

Determine whether there is enough information to reach a consensus. See "Assessing the Information Collected" (➥page 9-8) and "Findings" (➥page 9-13). Consider each project investigated individually first. Then, if there is a consensus about the topic for all projects, determine if there is a consensus for the organization as a whole. Record the results on the Validation Worksheet. See "Completing the Validation Worksheet" (➥page 9-10).

For each subprocess area where there is a consensus, try to develop finding statements. See "Findings" (➥page 9-13). The findings are abstractions of the information collected, expressed in terms of strengths, weaknesses, and improvement activities.

Next, determine if there is enough objective evidence to support each finding. All judgements made by the team should be correlated by at least two separate pieces of information. As the significance of the judgement increases, the correlation may require three or more separate sources of information. As a rule, if there is any doubt at all about whether a finding is valid, the team should defer it and should look for additional data during the consolidation interviews and final document reviews.

If the team does not have sufficient objective evidence to support a finding, it becomes a *candidate* finding. Candidate findings become the subjects of consolidation interviews or subsequent document reviews.

Once a preliminary finding has been made, the subprocess area is dropped from further consideration. That does not mean that new evidence will not be considered, but rather that the team will not spend any more time looking for data relative to the issue. This is necessary to allow the team to use its time on site most effectively.

If the team identifies a possible weakness, the development organization (through the site visit coordinator or in subsequent interviews) should be given an opportunity to produce evidence that might mitigate or eliminate the weakness. By double checking, the team avoids making findings based on anomalous responses. No direct mention is made of the preliminary finding. The request for clarification should define the subject matter and ask whether what the team observed or heard is representative. For example, the team might ask "We were not able to determine if the estimates for project size were based on actual data. Did we miss something?"

Step 19   Create Consolidation Plan

Objective   Plan and initiate further data collection.

Description   This step is special purpose team caucus intended to focus
the last part of the site visit on collecting the specific
information needed. The team decides what data or further
objective evidence it needs to finalize the candidate
findings, and plans how they will gather the information.
The team then initiates the next round of interviews and/or
document review. All interviews and document reviews
must be completed within the remaining time of the site
visit.

General guidance for conducting an effective team caucus
is provided in Chapter 9. Specific guidance which is
applicable to this step is provided in this section. The topics
most applicable to this step are listed below.

Chapter 6	Guidelines, Setting Priorities for the Consolidation Interviews and the Final Document Reviews	page 6-28
Chapter 9	Section 1, Team Member Roles	page 9-2
	Section 2, Consensus Process	page 9-5
	Section 3, Adjusting the Site Visit Plan	page 9-6
	Section 4, Assessing the Information Collected	page 9-8
	Section 5, Findings	page 9-13

Guidelines   **Setting Priorities for the Consolidation Interviews and the Final Document Reviews**

The consolidation interview and final document review should focus on resolving any issues that are still open.

Use the Validation Worksheets to identify the investigation topics that still need to be resolved. These include topics for which the team has not reached a consensus yet or topics for which there is a candidate finding that needs additional objective evidence to support it.

For each subprocess area and topic, determine what information is still needed to reach a consensus:

- Has the documentation for the subprocess area been identified?

- Are there still issues or questions about the process?

- Is the audit trail adequate to show that the process is being followed?

Identify the interview candidates that can answer the questions about the process and that can identify any additional documentation needed.

Use document review Checklist B and Checklist C generated earlier in the site visit to identify the specific documentation that will be needed (➡page 8-14).

If the team still has not determined what documents are needed to resolve the open issues, provide the site visit coordinator with general description of the type of information needed.

Determine the relative priority of the issues and how much time should be allocated to each issue before starting the final document review.

If further interviews are needed, prepare new Interview
Worksheets and coordinate the interview schedule with the
development organization's site visit coordinator. If
additional documentation is needed, coordinate the
requests with the site visit coordinator.

Step 20   Conduct Consolidation Interviews

Objective   Clarify any remaining issues by confirming or negating
                    candidate findings through further interviews.

Description   The consolidation interviews give the team a chance to
                    clarify any remaining issues and to identify any additional
                    documents that may be required to substantiate a finding.
                    The consolidation interviews use the Interview Worksheets
                    prepared in Step 19, "Create Consolidation Plan" (➡page
                    6-27).

                    The main difference between consolidation interviews and
                    exploratory interviews is in the amount of information the
                    team already has to guide it through consolidation.
                    Consolidation interviews usually focus on one or two
                    questions and are designed identify a specific piece of
                    objective evidence or resolving an issue that remains open
                    after the exploratory interviews and the document reviews.

                    General guidance for conducting interviews is provided in
                    Chapter 7. The topics most applicable to this step are listed
                    below.

Chapter 7	Section 1, The Interview Environment	page 7-2
	Section 2, Team Member Roles	page 7-5
	Section 6, Controlling the Interview	page 7-17
	Section 3, Selecting Interview Candidates	page 7-8
	Section 5, Developing Interview Questions	page 7-15
	Section 7, How to Record Results	page 7-19

Step 21   Conduct Final Document Review

Objective

Clarify any remaining issues by confirming or negating candidate findings through further document review.

Description

Final document review focuses on locating a specific piece of information that the team needs to confirm a candidate finding. The final document review follows the same techniques used in earlier document reviews except that the objective is more narrowly focused. This is the last opportunity to identify the objective evidence to support a candidate finding or to resolve any open issues.

The documents to be reviewed and the objectives for the document review are determined inStep 19, "Create Consolidation Plan" (➥page 6-27).

Do not spend a lot of time trying to locate a particular piece of information. Ask the site coordinator to identify the location of the information. Use a general description of the information needed. Do not describe the finding you are trying to verify.

General guidance for conducting document reviews is provided in Chapter 8. The topics most applicable to this step are listed below.

Section 3   Findings (Phase 5)

Objective   Consolidate the decisions made during the Site Data Collection phase.

Participant   SCE Team Members

Time   1/2 day

Entry Criteria

❑ The site visit (Phase 4) has been completed. The following outputs of Phase 4 are used in this phase:	
❑ Completed Validation Worksheets	page 6-20
❑ Completed Interview Worksheets	
(Exploratory Interviews) (Consolidation Interviews)	page 6-18, page 6-30
❑ Preliminary Findings	page 6-24
❑ Document Review Working Notes	
(Initial Document Review) (Document Review) (Final Document Review)	page 6-15, page 6-24, page 6-31

Activities

❑ Determine Findings (Step 22)	page 6-34
❑ Produce Findings Report (Step 23)	page 6-36
❑ Conduct Exit Briefing (Step 24)	page 6-38

Step 22   Determine Findings

Objective

Validate the preliminary findings and consolidate them by KPA.

Description

In a final series of caucuses, the team analyzes the information learned from the consolidation interviews and final document review to determine the official findings for the evaluation. The process is similar to the way the preliminary findings were developed in Step 18 (➥page 6-24).

Review the unresolved topics listed on the Validation Worksheets and the new information that was collected. If the team can reach a consensus about a subprocess area, develop candidate findings for the topics involved.

Review the preliminary findings and candidate findings, and determine if there is enough objective evidence to support a finding. In the absence of objective evidence the development organization should have the benefit of the doubt.

The validated preliminary findings become final findings, while the negated findings are dropped from consideration.

Group the findings by KPA. Review the findings until there is a consensus about how the findings are worded.

General guidance for conducting an effective team caucus is provided in Chapter 9. The topics most applicable to this step are listed below.

Chapter 9	Section 1, Team Member Roles	page 9-2
	Section 2, Consensus Process	page 9-5
	Section 3, Adjusting the Site Visit Plan	page 9-6
	Section 4, Assessing the Information Collected	page 9-8
	Section 5, Findings	page 9-13

Step 23   Produce Findings Report

Objective   Document the SCE activities and provide a formal record of
                the findings.

Description   The findings report is the formal record of the findings of
                the SCE along with any supporting information to show
                what was investigated and the objective evidence on which
                the findings were based.

                When the results of multiple SCEs will be compared, such
                as when SCE is used for source selection, the findings
                report should have the same style and format for all SCE
                site visits.

                The final report should include:

                1. **Information common to all development organizations**, including the Target Product Profile, the Target Process Capability, the Critical Subprocess Area List, etc.

                2. **Information provided by the individual development organization**, including Project Profiles, the Proposed Project Profile, organization charts and information, and responses to the Maturity Questionnaire.

                3. **All worksheets**, including Key Issue Worksheet, Validation Worksheet, and Interview Worksheets.

                4. **Objective evidence** that serves as a basis for findings. (This section should be a formal description of the evidence supporting the team's findings rather than the actual evidence. The team will not be allowed to take the evidence with them.)

                5. **Findings**, including a separate sheet(s) for each KPA. The findings should include references to the objective evidence that support them.

                Some portions of the report are generated during the visit,
                such as the findings. For accuracy, the remainder of the
                report should be generated as soon as possible after the site
                visit.

Chapter 6   Conducting the Site Visit
Section 3   Findings (Phase 5)
Step 23   Produce Findings Report

In most cases, the conclusion of the site visit represents the conclusion of the SCE team's activities.

In a source selection, the findings report must be complete enough so that sponsoring organization officials can understand all judgements made by the SCE team in case the SCE team is not available to explain them. In contract monitoring, the report must be complete enough so they can be compared to subsequent evaluations in a meaningful way.

Step 24   Conduct Exit Briefing

Objective

Provide feedback to the recipient and conclude the SCE.

Description

The exit briefing is the official end to the site visit. The content of the exit briefing depends on the purpose of the SCE and on the policy of the sponsoring organization, particularly when SCE is being used in a source selection.

At the very least the team should formally thank the organization for their support and let them know that the site visit has been concluded.

When allowed by the sponsoring agency, the exit briefing should include a presentation of the findings. If the team can not present the findings, they should tell the development organization if and when the findings will be made available to them and how to request the results.

In a source selection, the Procuring Contracting Officer (PCO) must agree to the agenda of the exit briefing. The acquisition process is controlled by regulations that puts severe constraints on "discussions" with development organizations. The PCO may decide that a debriefing of findings in any form constitutes a discussion. Customer feedback indicates that most source selection authorities are reluctant to let the SCE team present findings before contract award.

If the findings are presented at the exit briefing, do not get involved in a debate about whether the findings are correct. Let the development organization know that the findings are being presented as a courtesy.

The exit briefing should take about 15 to 30 minutes. If findings are presented, allow at least 1 hour.

Chapter 7　　Interviews

Chapter 7   Interviews

This chapter provides general information and guidance on conducting interviews. The information is most applicable to the following steps in the method:

- Step 11, "Prepare for Exploratory Interviews" (➡page 5-44)

- Step 15, "Conduct Exploratory Interviews" (➡page 6-18)

- Step 19, "Create Consolidation Plan" (➡page 6-27)

- Step 20, "Conduct Consolidation Interviews" (➡page 6-30)

Section 1   The Interview Environment

The effectiveness of an interview can depend on the environment in which the interview is conducted. If the people being interviewed are made to feel comfortable they will be more likely to respond freely and provide the required information. If they feel threatened by the interview environment, they will be more likely to hold back information or to tell you what they think you want to hear.

People who have been interviewed will talk about their experiences with those still scheduled to be interviewed. If the team conducts the interviews in a professional and non-threatening manner, it will have a positive effect on the later interviews.

Also, the interviews are the most visible part of the SCE to the organization being evaluated. The way the interviews are conducted will influence how receptive the organization will be to the results of the SCE. It may also influence how the organization reacts to future SCEs.

Creating a Productive Atmosphere

It is important to take a few minutes at the start of each interview to make the interviewee feel comfortable.

☞ Introduce the members of the team. This will make the interview seem less impersonal.

☞ Explain briefly how the interview will be conducted so that the interviewee will know what to expect.

☞ Explain that the SCE team will treat all that they hear as confidential so that information collected can not be traced back to a person or a project.

☞ Explain that it may be necessary to interrupt a response when the team feels they have heard the answer. This is done in the interest of saving time.

☞ Explain that the team will not always know who the appropriate person is to ask about a particular topic. The interview candidates should feel comfortable saying "I do not know" or "I am not the right person to ask."

☞ Let the interviewee know that there will be time at the end of the interview if they would like to add to or clarify anything that was said.

☞ Start the interview by asking the person to briefly describe their role in the organization.

☞ During the interview, follow the plan.

- Cover the topics in some logical order. Do not jump back and forth between subjects.

- Have one person ask the planned questions.

- Coordinate follow-up question through the person leading the interview.

☞ At the end of the interview, ask if there is anything else that the team should know. Ask "Are you comfortable that you have said everything you want to?"

Room Layout

The layout of the interview room is an important factor in the interview environment. The fact that the interviewee is being interviewed by a team can be intimidating. This effect can be exaggerated if the team is lined up on one side of the room and the interviewee is sitting alone on the other side.

A table can help to make everyone feel more comfortable. The table should be large enough that everyone can sit comfortably around it, but not so large that it makes people feel separated.

Many teams prefer to sit in a circular arrangement. A circle tends to make the interviewee feel included in the group rather than opposed to it. Even when a rectangular table is

used, a circular feel can be created by having the team members sit around the ends of the table instead of along one side.

Nonverbal Factors

Communication depends on more than just what people say. Nonverbal factors are also an important part. They can support what is said or they can send a very different message.

Try be aware of signals you send to the interviewee. Body language is particularly important. You should appear to be attentive, interested, and non-judgemental. If you seem to be intimidating, unreceptive, or disapproving, you could have an undesirable effect on what is said.

On the other hand, try to avoid being influenced by nonverbal factors from the interviewee. Body language is easily misinterpreted. It should not color your judgement about what is said.

Non-attribution

Non-attribution is an important factor in the SCE interview process. People generally do not feel they can talk freely if they think that what they say will get back to their organization.

Conduct the interviews with one person at a time. At the beginning of each interview, explain that findings will be reported for the organization as a whole and that nothing said will be attributed to an individual or project.

Section 2   Team Member Roles

The team has a lot to investigate during the interview sessions. You will hear a lot of important information and you need to be sure that the pertinent data is accurately recorded. Assigning specific roles to team members during the interviews will help you manage the interviews more effectively.

Roles do not need to be assigned to the same people for every interview. Roles may be assigned by some type of rotation scheme or they may be assigned based on the topics to be investigated. Whatever scheme is used, it is important that everyone know their role at the start of each interview.

Interview Leader

The interview leader is responsible for making sure that the interview plan is followed.

Before the interview starts, the interview leader:

- Makes sure that everyone knows their role.

- Makes sure that everyone understands the objectives and schedule for the current interview.

At the start of the interview, the interview leader:

- Welcomes the interview candidate.

- Introduces the team members.

- Explains how the interview will be conducted ().

During the interview, the interview leader:

- Asks the questions on the Interview Worksheets and may ask follow-up questions to make sure that the team gets the information they are looking for.

- Keeps the session organized by coordinating follow-up questions from other team members.

- Keeps the responses focused when necessary, by politely interrupting when the question has been answered or when the response is not pertinent to the question.

At the end of the interview the interview leader:

- Asks the other team members if they have any other questions.

- Asks the interviewee if there is anything else that he or she would like to ad.

- Thanks the interviewee.

Time Keeper

The time keeper is responsible for keeping the interview cn schedule. The time keeper monitors the time throughout the interview, and lets the interview leader know if it looks as if the interview is taking longer than planned.

Recorder

The recorder keeps the official record of the interview. The recorder should take down what is said and should not editorialize. The recorder should focus on hearing and recording what is said and should not be analyzing what is said or thinking about follow-up questions to ask.

The interviewee's responses should be recorded on a copy of the Interview Worksheets. Any questions that were not on the Interview Worksheets should also be recorded.

Document Recorder

Some teams designate a person to keep a list of the objective evidence that the team hears about during the interview. At the end of the interview, the document recorder reads the list for the interviewee and the other team members to verify that the list is correct. The list will be given to the site visit coordinator (➡page 6-6) to provide the documents for the next document review.

If the team does not have a designated document recorder, this list is maintained by the recorder.

Others

Other team members are responsible for listening to the interviewee's responses and making sure that the team gets the information they are looking for. They should ask follow-up questions when necessary.

The other team members should record their own notes from the interviews.

Section 3   Selecting Interview Candidates

The team must decide who to interview and which topics to investigate with each interview candidate.

Use the organization charts submitted by the development organization to prepare the initial interview plan. The interview candidates should be identified by their position in the organization, not by name. The person may not be in the same position by the time of the site visit.

Individual organizations will have their own titles for positions and will have their own way of assigning roles and responsibilities. The plan may need to be adjusted as the team understands more about the roles and responsibilities of the positions during the site visit.

The following types of positions should be considered for the initial plan:

- **Senior managers** - those managers who have responsibility for defining policies and procedures that apply to all projects and for monitoring compliance with those policies and procedures. This includes managers of functional areas, such as quality assurance, that have responsibility across multiple projects.

- **Project managers and key practitioners** (CM, SQA, test director, subcontract, software) - the personnel responsible for defining the specific processes that should be followed on the project and for monitoring those processes.

- **Project first-line supervisors** - the personnel responsible for overseeing the day to day work practices on the project.

- **Project staff** - the individuals who perform the various activities of the organization's software processes.

The interview plan should be designed to gather the following information for each critical subprocess area:

☞ What do the people at various levels of the project understand the process to be?

- Those who establish the organization level policies and directives.

- Those who define the standards and procedures for the project.

- Those who execute the process.

- Those who verify that the process is followed.

☞ What documents will provide the objective evidence of the process?

- What are the organization policies that apply to the subprocess area?

- What are the project procedures and standards that apply to the subprocess area for each project?

- What documentation is produced that shows evidence that the processes are being followed?

☞ Who else should be interviewed for information the subprocess area?

- Who is responsible for creating, approving, executing, and monitoring each policy, procedure, or standard?

It is important to seek process information from all levels in an organization, both up and down the chain as well as across projects.

Plan to interview the managers first to establish what organization level policies and procedures are in place and to confirm who best to interview for subsequent interviews. Start with the lowest level of management that has responsibility for all projects being evaluated.

Appendix G, Section G.1 (☞page G-3) contains examples of agents, artifacts, and relationships which might be associated with the software development activities an organization performs. These examples are organized by subprocess area. The agents are the people associated with

the activities. They may perform the activities, they may
provide the input to or receive the output from the
activities, or they may control the activities. The examples
of agents may provide clues to possible interview
candidates for each of the critical subprocess areas.

Section 4   Mapping Topics to Interview Candidates

Each of the topics on the Validation Worksheet is allocated to a specific interview candidate. In general, more than one person may be interviewed about a single topic, and one person may be asked about multiple topics.

Creating a matrix of interview candidates and topics such as the one show in Figure 7-1 (➡page 7-12) can be helpful in developing an interview plan. The matrix can be used as a guide when developing Interview Worksheets for a specific candidate. It may be particularly useful when a candidate has a general knowledge of a wide area to make sure that all of the intended topics are covered. It shows:

1. Which KPAs each interview candidate is being interviewed about.

2. The interview candidate's role relative to the process.

3. Which projects the interview candidate has knowledge of.

Ideally, the site visit should probe each critical subprocess area at the organization level and for each specific project and at each level of responsibility for the processes that support the subprocess area. Not all of this information can be acquired through interviews because of the time constraints but it is important to know where the gaps in the interview plan are. A matrix of interview candidates and topics can show you where the gaps are so that you can ensure that the gaps are consistent with the investigation priorities and not due to an oversight.

If you are not sure who to talk to about a particular topic, plan to ask who the appropriate person is during interviews at the most senior level appropriate to the topic.

The following guidelines should be considered when allocating topics to interview candidates:

Interview Candidate Topic Matrix — Interview Candidates: Position (and Name)	Scope of Responsibility					Role				Key Process Area												
	Organization Level	Project A	Project B	Project C	Project D	Defines Policy	Defines Process	Executes Process	Monitors Process	Requirements Management	Software Project Planning	Software Project Tracking and Oversight	Software Subcontract Management	Software Quality Assurance	Software Configuration Management	Organization Process Focus	Organization Process Definition	Training Program	Integrated Software Management	Software Product Engineering	Intergroup Coordination	Peer Reviews
Division Manager	x					x					x	x				x	x	x			x	
Division SW Manager	x						x		x							x	x		x	x	x	
Division QA Manager	x					x			x					x								
Division Test Manager	x					x			x	x										x	x	
Process Group	x					x	x									x	x	x				
Proj A Program Mgr		x				x			x	x	x	x	x									
Proj A System Manager		x				x			x		x	x										
Proj A SW Manager		x				x			x	x	x	x	x			x	x	x	x	x	x	x
Proj A Test Manager		x				x			x	x										x	x	x
Proj A SW Group Leader		x						x	x	x	x	x				x	x	x	x	x	x	x
Proj A SW Engineer		x						x						x	x				x		x	x
Proj A CM Rep.		x					x	x							x							
Proj A QA Rep.		x					x	x						x								x

Figure 7-1: Interview Candidate Topic Matrix

☞ Spend the least amount of time at the senior levels, and the most time at the working level.

There is a natural tendency to spend a lot of time with the top level people because they should have the broadest perspective about what the organization's software processes are. However, they may not know the details about the organization's activities and there may be a big difference between the organization's policies and actual practice. The team will need time to interview the people who are doing the work to gather the objective evidence needed.

☞ Interviews with the organization level managers should concentrate on finding out:

- What are the organization level policies and procedures?

- What documentation will show evidence that the policies and procedures are being followed?

- Who in the project is responsible to carry out the policies and procedures?

☞ Interviews with project management should concentrate on finding out:

- What are the project level procedures and standards?

- What documentation will show evidence that the procedures and standards are being followed?

- Who on the project team is responsible to carry out the procedures and standards?

☞ In interviews with team members concentrate on finding out what processes are actually followed.

Appendix G, Section G.1 (☞page G-3) contains examples of agents, artifacts, and relationships which might be associated with the software development activities an organization performs. These examples are organized by subprocess area. The agents are the people associated with the activities. The artifacts are the work products which are part of the activities. The relationships indicate the roles the agents and artifacts play in the activities. The

examples of agents, artifacts, and relationships may
provide clues about subjects to investigate with each
interview candidate.

Section 5   Developing Interview Questions

The following guidelines are provided for developing interview questions.

☞ Each question should be related to one specific topic. A topic is a feature (➡page 2-9) associated with a subprocess area.

☞ When transforming the topic into a question, the team may consider the following ways to direct the focus of the question:

  • **Definition:** What policies, directives, standards, or procedures define the process and specify how it is to be carried out?

  • **Initialization:** What activities are required to initiate the process?

  • **Objective evidence of use:** What is the documentation produced to show evidence that the process is being followed?

  • **Visibility and control:** Who is responsible for monitoring that the process is being followed and how is that information reported up the management chain?

  • **Robustness under change:** How are changes to policies, standards, and procedures controlled?

☞ Determine what information is needed from the interviewee.

Does the question wording clearly probe for that information? In the following examples of probe questions, "X" represents a specific activity:

  • Please describe how you do "X"?

  • What are the steps?

  • What is the output?

  • Who is responsible for "X"?

- How does your organization verify that the process for "X" is being followed?

- What documentation is associated with "X"?

☞ Include questions that will show what documents need to be reviewed.

In particular, ask questions that will identify objective evidence that will show that processes are being followed.

☞ The questions should be open-ended but of reasonable scope so that one person can readily and succinctly answer in few sentences.

The questions should be at a level of abstraction that is low enough that the identified work practices may be carried out by one person or a small group.

- Say, "Please describe your process for..."

- Go from general description to specific practices

☞ Questions should not be structured for "yes" or "no" answers unless they are follow-up questions to confirm a particular answer that was heard.

☞ Questions should not be leading or judgemental.

If the question implies a correct answer, the interviewee may tell you what you want to hear. Likewise, if the question implies a wrong answer, the interviewee may try to tell you something different.

- Do not start with conclusion

- Do not say, "I assume..."

- Do not ask, "Does that make sense?"

- Do not enter into or start a discussion.

Section 6   Controlling the Interview

The SCE team needs to establish a comfortable interview environment where the interviewees feel as if they can speak freely. However, they also need to keep the interviews focused on the objectives of the site visit. The team needs to maintain control without losing the desired environment.

☞ Make sufficient preparations before each interview. An interview can not be any better than its preparation.

- Develop an Interview Worksheet for each interview.

- Assign specific roles to team members for each interview. Make sure everyone knows their role before the interview begins.

- Review the objectives before each interview and make any necessary changes to the Interview Worksheets.

☞ At the start of each interview, ask the interview candidates to describe their roles in the organization to verify that they are the right people to talk to about the selected topics.

☞ Do not continue to spend time with a particular topic if the interviewee does not know about it.

Ask who the appropriate person would be to talk to about it and move on to another subject.

☞ Keep follow-on questions relevant to the investigation topics.

Think about follow-up questions before asking them. The questions should be necessary to pursue the topic. Do not get side tracked and ask questions out of personal curiosity.

☞ If the interviewee's response is not pertinent to the question, politely interrupt and restate the question.

☞ If the team feels they have heard enough about a particular topic, politely interrupt and move on to the next topic.

Be careful not to make the interviewee feel as if they have not had a chance to explain everything they wanted to. Explain that the team believes that they have the information they need.

Section 7   How to Record Results

The SCE team creates an Interview Worksheet for each person they plan to interview. The worksheets are initially created in Step 12 during the Specific Preparation phase. The worksheets are updated and new ones are created throughout the site visit.

Each team member should have a copy of the Interview Worksheets for the current interview. Each person should record their own notes on their copy of the worksheets. One team member should be designated as the recorder for the interview. The recorders set of worksheets will be the official set for the interview.

Record the responses in the space provided after each question. Record any changes to the questions and any follow-up questions.

Record what is actually said. Do not editorialize.

After the interview, hold a brief team caucus while the information is fresh in everyone's minds. Review the notes from the interview to get concurrence that the important points have been captured and that the wording is accurate.

Chapter 8   Document Review

Chapter 8   Document Review

This chapter provides general information and guidance on document reviews. The information is most applicable to the following steps in the method:

- Step 14, "Conduct Initial Document Review" (➡page 6-15)

- Step 18, "Develop Preliminary Findings" (➡page 6-24)

- Step 19, "Create Consolidation Plan" (➡page 6-27)

- Step 21, "Conduct Final Document Review" (➡page 6-31)

Section 1   Types of Documents

The SCE method looks at documents in three broad categories:

- Organization Level Policies and Procedures.

- Project Level Plans, Standards, and Procedures.

- Evidence of Process Activity.

**Organization Level
Policies and Procedures**

Organization level documents are established by senior management and are applicable across multiple projects. They help to provide a framework for consistency among projects and a basis for organization wide process improvement.

Organizational policies communicate the goals of the organization to individual project managers. They define what the organization needs and expects in relation to the project work.

Organization level procedures document the standard way of developing software products within the organization. These procedures may be detailed enough that they could be followed as a default way of doing business. They may be tailored to the needs of the individual project in accordance with prescribed rules.

In the absence of complete procedures at the organization level, the organization may provide guidelines or templates for projects to follow to develop their own procedures.

**Project Plans,
Standards and
Procedures**

Project level documents define the specific processes to be followed for the project. They are usually developed by key project personnel in accordance with any applicable

organization level policies and procedures. Project specific standards and procedures may need to follow standards imposed by the sponsoring agency.

Project plans are the basis for managing and tracking project activities. Standards are used to enforce consistency in the software development process. Procedures describe how the process is to be performed.

Evidence of Process Activity

Evidence of process activity refers to the track record that is usually left behind from the work that is performed. It is something tangible that shows that the activities described are actually taking place.

Evidence of process activity will generally include types of information that are normally not thought of as documents. It may consist of such things as:

- schedules,
- memoranda,
- meeting agendas and minutes,
- action items,
- status reports,
- deficiency reports,
- unit development folders, and
- database reports.

Section 2   What to Review

The SCE team needs to identify the documents that will provide the objective evidence to show what the development organization's software development processes are and to show evidence that the processes are followed. Throughout the site visit, look for clues to which documents will provide that information.

The set of critical subprocess areas and topics listed on the Validation Worksheets provides the basis for the site visit investigations. All document reviews must relate back to those investigation topics.

The development organization's presentation during the initial organization meeting (➡page 6-13) may provide some information about what documents to look at. Listen for such information as:

- The names of key organizational policy documents.
- Organization level procedure manuals.
- Project level plans, standards, and procedures.
- Standard reports.

Document review starts with a set of documents that the development organization makes available for the initial document review. This set of documents is usually requested in terms of a general description of what the team wants to see (➡page 6-7). Select those documents that are most relevant to the investigation topics on the Validation Worksheets out of all the documents the organization may provide.

While going through the initial document review and exploratory interviews, look for clues to other objective evidence.

Try to identify the documents needed and request them from the site visit coordinator (➡page 6-6) as early in the site visit as possible. Request only a few samples of each type of document to keep the document requests manageable. Request documents for a specific time period as a check that the documentation was not maintained sporadically.

When specifying the time frame for samples of audit trail data, take the project schedule into consideration. Ask for data from a period in time when the process that generates the data is well under way. For example, don't ask for integration test reports from a month that is prior to the start of integration testing.

When requesting audit trail data, specify samples from periods of time that will show continuity of the process used. This could mean asking for examples that span several months.

Appendix G, Section G.1 (➡page G-3) contains examples of agents, artifacts, and relationships which might be associated with the software development activities an organization performs. These examples are organized by subprocess area. The examples of artifacts may provide clues about documents that the team could expect to find for each of the critical subprocess areas.

Section 3   What to Look For

When reviewing documents, look for the definitions of the processes the organization follows and evidence that the processes are being followed. Do not evaluate the technical content of the documents.

For example, a you should not try to evaluate whether the documented design is valid to satisfy the requirements. Instead, look for evidence of processes to develop, maintain, document, and verify the design.

The following are general types of information that the teams need to collect during the document reviews:

- What are the documents that correspond to the critical subprocess areas?

  Identify the documents that are pertinent to the topics they are investigating. For each topic, look at documents in all three categories: organization level, project level, and evidence of process activity.

- What is the hierarchy of requirements among the documents?

  Verify that the documents produced as part of the process satisfy the process requirements. Do project specific documents comply with organization level policies and directives? Do audit trail documents comply with organization and project level requirements?

  You are limited in how much you can verify that a document satisfies the requirements. You can verify that the right types of information are included but you may not be able to verify the content. If an approval process is specified for the document, you can look for evidence that the process was followed.

- Does the audit trail information verify that the documented processes are being followed?

- Do the processes defined in the documents satisfy the CMM goals corresponding to the applicable critical subprocess areas?

Documents that define the processes for the organization or for a specific project need to be managed. The scope and authority of these documents need to be clearly understood within the organization and changes must be controlled. Checklist A (➡page 8-14) may be used as a guide to the types of information that should be provided for these documents.

The following sections provide additional guidelines on what to look for when reviewing documents in each of the three categories: organization level, project specific, and evidence of process activity

Chapter 8   Document Review
Section 3   What to Look For
Guidelines   Reviewing Organization Level Policies and Directives

Guidelines   **Reviewing Organization Level Policies and Directives**

☞ When reviewing organization level documents, have the following goals in mind:

     a. To identify the organizational policies and standards that control the software development process.

     b. To evaluate the extent to which processes are institutionalized and supported at the organization level.

     c. To extract information that may be used to refine the plans and schedule for the remainder of the site visit. (i.e., What documents should be reviewed? Who should be interviewed?)

☞ Look for information for evaluating the policy and directive system:

- Is a directive system for the organization's policies and directives visible to project staff?

- Is there a configuration management responsibility for these policies and directives?

- If documents are in electronic format, what is the policy for control, approval and release of electronic format documents?

- Have all of the organization level policies and directives pertinent to the critical subprocess areas been identified?

- Are there any subprocess areas which are being investigated that are not supported by organization level policies and directives?

- Do the documents contain the appropriate "look for" items for that type of document?

- Is there an adequate organizational "commitment to perform" and "ability to perform" for each for the critical subprocess areas?

☞ Look for clues about what other documents to request:

- Are there standard procedures which will be tailored for each specific project or is there a requirement for how the project specific procedures should be developed?

- Is guidance provided on how procedures should be tailored?

- How do the organization level documents relate to the critical subprocess areas?

- What project specific standards and procedures are specified in the policies and directives? (What project specific documents should be requested for review?)

- How do the project specific standards and procedures relate to the critical subprocess areas and organization level policies and directives?

- What reporting and controlling mechanisms exist between organization level management and individual projects? (What documents or paperwork will show evidence of activities being performed, monitored, and controlled)

☞ Look for information that may be used to refine the interview plans:

- What are the roles and responsibilities of organization level positions and of key project specific positions?

- Who defines organizational policies and directives?

- Who develops project specific plans, procedures, and standards?

- What is the approval process for organization level documents and project specific documents?

- What are the controlling and reporting relationships among organization level positions and key project specific positions?

- How do the various roles and responsibilities relate to critical subprocess areas?

Guidelines   **Reviewing Project Standards and Procedures**

☞ When reviewing project standards and procedures, have the following goals in mind:

   a. To identify for each project investigated the documents that define the specific processes that are pertinent to the critical subprocess areas.

   b. To determine if projects conform to organizational policies and standards.

   c. To evaluate whether the defined processes are valid for controlling the product development.

   d. To extract information that may be used to refine the plans and schedule for the remainder of the site visit. (i.e., What other documents should be reviewed? Who should be interviewed?)

☞ Look for information that will show whether projects conform to organizational policies and standards:

   • Do the project specific plans, procedures, and standards specified in the organization level policies and directives exits?

   • Are the project specific documents tailored versions of the organization level documents? Do they comply with the organization level requirements?

   • For any documents that do not comply, is there an explicit noncompliance approval?

   • Is there a record of organization level review and approval of project documents?

   • Are they approved at the appropriate level?

☞ Look for information for evaluating the defined processes used on each project:

   • Is the system of plans, standards, and procedures visible to project staff?

   • Is there a configuration management responsibility for these documents?

- Have all of the project level plans, standards, and procedures applicable to the critical subprocess areas been identified?

- Are there any subprocess areas which are being investigated that are not supported by written standards and procedures?

- Do the documents contain the appropriate "look for" items for that type of document?

- Are the defined processes adequate for controlling the product development in each of the critical subprocess areas?

☞ Look for information for completing the document review plans:

- How do the project specific plans, standards, and procedures relate to the critical subprocess areas and organization level policies and directives?

- What reporting and controlling mechanisms exist between project staff and project leaders? (What documents or paperwork will show evidence of activities being performed, monitored, and controlled)

☞ Look for information that may be used to refine the interview plans:

- What are the roles and responsibilities of key project level positions?

- Who develops project specific plans, procedures, and standards? What is the approval process for project specific documents?

- What are the controlling and reporting relationships among project staff and project leaders?

- How do the various roles and responsibilities relate to critical subprocess areas?

Guidelines   **Reviewing Evidence of Process Activity**

☞ When reviewing evidence of process activity, have the
   following goals in mind:

   a. To collect objective evidence to validate what the
      team has heard and read about the
      organization's software process capability.

   b. To determine if project work that is pertinent to
      the critical subprocess areas is being carried out
      in a manner consistent with project plans and
      procedures.

☞ Look for information to show whether the audit trail
   documentation complies with organization level and
   project requirements:

   • Do all of the types of audit trail documents which
     were identified through the document reviews
     and interviews exist?

   • Do they have the format and content required by
     organization level or project requirements?

   • Do the documents have the distribution list
     required by organization level or project
     requirements?

   • Is there a record that reports were received and
     acted upon?

   • Are periodic reports generated at the frequency
     required by organization level or project
     requirements? Are activity reports or deficiency
     reports generated within the required time?

   • Do the documents contain the appropriate "look
     for" items for that type of document?

☞ Look for objective evidence to validate that the
   processes you have heard and read about are actually
   being followed:

   • How do the audit trail documents relate to
     organization and project level documents and to
     critical subprocess areas?

- Are there any critical subprocess areas for which an audit trail has not been found?

- Is the audit trail adequate to show that the defined processes are being followed?

- Does the audit trail show that organization and project level management adequately monitor and control the process?

Section 4   How to Record Results

The SCE method provides three types of forms for recording initial document review results:

1. Checklist A is used to identify what organization and project level support documents exist to guide project work. The data listed in Checklist A indicates how the document is used and controlled within the organization.

2. Checklist B is used to map each of the organization and project level support documents for which Checklist A has been completed to the critical subprocess areas that the team will investigate.

3. Checklist C is used summarize by KPA what the team has found regarding organization and project level documents to support project work.

Checklist A

Checklist A is shown in Figure 8-1 (➥page 8-15). This checklist should be completed for each document that plays a role in defining the organization's processes. These include policies, directives, standards, instructions, and procedures. They include both organization level and project specific documents.

The purpose of Checklist A is to make a record of the organization and project level support documents (such as policies and procedures) that are available to guide the project work. The checklist helps the team to see how the documents are controlled and allows the team to judge how the structure of the set of documents supports the project work.

The type of information listed on Checklist A is generally contained within in the document itself, but some organizations may handle that information differently. For example, information such as the scope of the document could be specified in a policy statement or master plan that

Checklist A
(for each policy and procedure)

Name: ☐ Policy

                                               ☐ Procedure

Information sought	Found	Comment
1. Source quality Who originated? Is a review record visible? Is the approval record visible? Is the audience identified? Is the date it came in force visible?		
2. Version data Does it have a current version number? Is the version number it supersedes visible?		
3. Content Is a scope defined? Is there a purpose statement?		
4. Glossary Are there definitions of terms and/or acronyms?		
5. Relationships If policy, is there a list of procedures identified? If procedure, is there a defined tailoring mechanism for projects to use?		
6. Configuration management control Is there a configuration management responsibility for these policies and procedures?		

Figure 8-1: Document Review Checklist A

is more conveniently revised. That way a document originally developed for a specific pilot project could be applied to more projects without modifying the original document. As another example, the configuration management responsibility for organization and project level documents may be specified in a separate policy or procedure rather than in each document. What is important is that the information is available and visible to the project members.

The checklist is to be used as a guide and should not be used as criteria for evaluating the document. If some of the information is not available in a document, consider the purpose of the document and evaluate how the lack of information affects the usefulness of the document.

The checklist does not address the content of the document. Only those documents that apply to the critical subprocess areas should be evaluated, and this evaluation can only be in terms of the extent rather than goodness. Extent refers to whether the document contains the full range of coverage of the processes necessary to satisfy the CMM goals for the critical subprocess areas. The common features are indications of the extent needed.

The choice of which documents apply to the critical subprocess areas is the focus for Checklist B.

Checklist B

Checklist B is shown in Figure 8-1 (➡page 8-15). This checklist is used to connect each of the organization and project level documents, for which Checklist A was completed, with the selected critical subprocess areas that are the scope of the investigations.

Each row in Column 1 in Checklist B is a critical subprocess area.

Checklist B
Organization's Policies and Procedures
Related to Critical Subprocess Areas

Critical Subprocess Areas	Applicable Organization Documents Plus Comments

Figure 8-2: Document Review Checklist B

During the initial document review the team identifies which of the Checklist A documents applies to which selected critical subprocess areas. The team also notes the lack of extent by which these applicable documents support the project work.

Note that the mapping of the organization structure to the CMM standard enables the team to understand how the organization thinks about software process support for its project work.

Checklist C

Checklist C is shown in Figure 8-1 (➡page 8-15). This checklist is used to summarize by KPA the support given to the project work by the documents applicable to the critical subprocess areas. It helps locate the objective information and provides a basis by which the team can roll the total document review results up into findings.

The example shown includes the list of KPAs for the *defined* and *repeatable* maturity levels. Create your own list consisting of the KPAs in the Target Process Capability.

Additional Notes

You may keep additional notes to record information to discuss at the team caucus. This information may include:

- Other documents to request.

- Changes to the interview plan.

- Candidate findings.

Checklist C
Summary of Organizational Documents by Key Process Area

Key Process Area	Comment
Requirements Management	
Software Project Planning	
Software Project Tracking and Oversight	
Software Subcontract Management	
Software Configuration Management	
Software Quality Assurance	
Organization Process Focus	
Organization Process Definition	
Software Product Engineering	
Integrated Software Management	
Intergroup Coordination	
Peer reviews	
Training Program	

Figure 8-3: Document Review Checklist C

Chapter 9　　Team Caucus

Chapter 9   Team Caucus

This chapter provides general information and guidance on conducting a team caucus. The information is most applicable to the following steps in the method:

- Step 16, "Hold Team Caucus" (➡page 6-20)

- Step 18, "Develop Preliminary Findings" (➡page 6-24)

- Step 19, "Create Consolidation Plan" (➡page 6-27)

- Step 22, "Determine Findings" (➡page 6-34)

Section 1   Team Member Roles

The team caucus is a highly specialized form of a team meeting. The team needs to review and analyze a large amount of data and make decisions about that data in a short amount of time. Applying some principles of meeting management can help keep the team focused on the task at hand.

Assigning specific roles to the team members helps to make efficient use of time. Without assigned roles, the team members tend to become involved in the details of the discussion. They may not realize that the time is not being used effectively. Also, important points may not be recorded. The assigned roles help to ensure that someone is thinking about the operation of the caucus.

Roles do not need to be assigned to the same people for every team caucus. Assignments may be rotated so that each person gets a chance to participate in the discussions. If possible, the people who are the most likely to be involved in the current discussion topic should not be assigned a specific role so that they are free to participate.

Facilitator

The primary function of the facilitator is to keep the caucus focused on the current task. The facilitator should concentrate on how the caucus is operating rather than on the details of the discussion.

At the beginning of the caucus, the facilitator should:

- Review the role assignments for this caucus.

- Review what needs to be accomplished.

- Assign a block of time for each task.

During the discussion the facilitator should:

- Make sure that the discussion is focused on the current task.

- Observe how team members are interacting and step in when there are problems.

- Look for indications that more data is needed for a consensus and focus the discussion on what additional information needs to be collected.

- Help the team move toward a consensus.

Recorder

The recorder is responsible for keeping the official set of notes from the meeting. The recorder should be listening and not participating in the discussion.

During the caucus the recorder should:

- Record decisions on the Validation Worksheets.

- Make a list of any documents to be requested.

- Record any changes to Interview Worksheets or plans.

- Record any preliminary findings or issues to be resolved.

At the end of the caucus the recorder should review the information recorded and give the team members a chance to make changes.

Time Keeper

The time keeper is responsible for watching the clock during the caucus and for letting the team know when the time allocated to the current task is nearly up. The team must then decide to allocate more time and adjust the schedule accordingly or to move on to the next task and schedule another time to finish the current task.

Others

Other team members are responsible for accomplishing the objectives of the current task. They should try to keep their own notes even though there is a designated recorder. They should also assist the facilitator in keeping the caucus focused on the objectives of the SCE.

Section 2   Consensus Process

The SCE team is not an autocracy where the leader dictates what decisions are made. Nor is it a democracy where the team votes and the majority prevails. Instead, the decisions are reached by team consensus.

Consensus is achieved when team members openly discuss the issues and work together to reach a decision that all of the members can accept. Team members must feel that they have had a chance to express their views and that their views have been heard.

Team members may not feel that the decision reached by consensus is the best choice but they should feel that it is a valid choice. Each team member should feel that they can support the decision because it was reached openly and fairly.

A decision by consensus has the following characteristics:

- Common basis of understanding.
- General agreement.
- No strong minority dissension.
- Retains group integrity and mission focus.

Reaching a decision by consensus requires the following steps:

- Sufficient analysis of data.
- Freely shared, open discussion of viewpoints on the data.
- Assignment of rank to issues.
- Focused discussion on possible decisions for dealing with issues.

Section 3   Adjusting the Site Visit Plan

The initial plans for the site visit are generally based on little information about the development organization. As you conduct exploratory interviews and document reviews you will collect information that will help you to refine the plans.

In each team caucus, consider the data most recently collected and think about the following:

☞ What information did the team expect to find in the last document review or interview session that was not found?

☞ Are there other interviews or document reviews planned which may provide the information needed?

☞ Do the Interview Worksheets contain the right questions to probe for the data or are new questions needed?

☞ Has enough information been collected about a topic to generate a finding?

☞ Are there other interviews or document reviews planned which are no longer needed or which may be scaled back?

☞ Is there someone else the team should interview?

☞ Is there another document the team should review?

Make the necessary changes to the site visit plans:

• Generate new Interview Worksheets or modify existing Interview Worksheets.

• Modify the interview schedule and make the necessary arrangements with the site visit coordinator (➡page 6-6).

• Request additional documents.

Spend a portion of time in each team caucus reviewing the plan for upcoming activities to see if any changes are needed. At a minimum, review the objectives of the next scheduled interview to focus the team on what to look for.

Section 4   Assessing the Information Collected

Frequently review the information collected throughout the site visit to determine if you have enough information to reach a consensus about the topics listed on the Validation Worksheets (➡page 5-42). If not, determine what additional information is needed.

A key assumption in SCE is that the development organization is motivated to put its best foot forward. You must have objective evidence to verify that the processes you hear and read about are actually followed.

All judgements made by the team should be correlated by at least two separate pieces of information. As the significance of the judgement increases the correlation should be from more separate pieces of information, with the most critical judgements substantiated by documented evidence.

Screen the information collected against the list of critical subprocess areas. If the information is not applicable to the critical subprocess areas it is outside the scope of the investigation.

Appendix G contains probing guides to help the teams evaluate the data collected. Section G.2 (➡page G-29) contains probing guides which are applicable to all subprocess areas. Section G.3 (➡page G-36) contains probing guides which are specific to each subprocess area. Refer to the instructions at the beginning of these sections for information on how to use the probing guides.

Consolidate all of the available information about the topics pertaining to a subprocess area. The Interview Worksheets (➡page 5-50) contain the subprocess area names to help consolidate questions and responses. Document Review Checklist C (➡page 8-18) can be used to help consolidate the document review responses by KPA.

Evaluate the information for each topic with respect to each project for which the topic was investigated. Use the collective professional judgement of the team and a consensus decision making process to determine if the goal associated with the subprocess area is satisfied with respect to the topics investigated. If the team reaches a consensus about whether the goal is satisfied, record the consensus on the Validation Worksheet (➡page 9-10).

Organization level conclusions about a topic on the Validation Worksheet are generated by consolidating the conclusions found for each project.

- If all projects satisfy the subprocess area relative to the topics probed, evaluate the likelihood that a new project will either follow the same processes or make planned improvements. If you feel that organization level procedures are adequately defined and regulated then the topic area is satisfied at the organization level.

- If no project satisfies the subprocess area relative to the topic, then the subprocess area is not satisfied at the organization level.

- If some of the projects satisfy the subprocess area relative to the topics but others do not, then you must look at the reason for the difference. The organization may have either adopted new procedures that do not apply to older projects or modified the process for a specific project to meet the requirements of its customer. You must evaluate the likelihood that the new project will use the defined processes.

Exercise judgement in interpreting what you see. Evaluate the processes of the organization in terms of their ability to reduce or control risk in the new development:

- The organization adds risk to the acquisition program if the goals of the Target Process Capability KPAs are not met

- The organization reduces risk to the acquisition program if activities are performed which are above and beyond the goals of the KPAs in the Target Process Capability

Instructions   **Completing the Validation Worksheet**

Use the Validation Worksheets to record the decisions
made during a team caucus. The worksheets were created
in Step 6. The investigation topics were added in Step 10.

There is one Validation Worksheet for each subprocess
area investigated. Each worksheet contains a section for
each topic to be investigated for the subprocess area. The
topic section is divided into 6 columns. Column 1 contains
the name of the topic. Columns 2 through 5 are used to
record the team's consensus about the topic for a specific
project. These columns are divided into rows with one row
for each project. Column 6 is used to record the team's
consensus about the topic for the organization as a whole.

An SCE team may reach a consensus about a topic for a
single project at any time during the site visit. Enter the
consensus on the worksheet as follows:

- Select the row within the topic section for the project to
  which the consensus applies.

- Select the **Explore Interview, Doc Review** or
  Consolid Interview column to indicate the source of
  the information on which the consensus is based.

- Place a "**Y**" on the validation form if the KPA goal is
  satisfied with respect to the topic and "**N**" if the goal is
  not satisfied.

If the team has not reached a consensus about a topic for a
single project by the end of the exploratory interviews,
enter "?" in the **Explore Interview** column to indicate
that this topic should be included in the consolidation plan.

If the team has not found the objective evidence for a topic
during the document reviews, enter "?" in the **Doc Review**
column to indicate that this topic should be included in the
consolidation plan.

When the team reaches a consensus about a topic for the organization as a whole, enter the result in the **Organization** column.

1 Use the set of the Validation Worksheets updated in Step 10.

2 When the team reaches a consensus about a topic for a specific project enter "**Y**" or "**N**" to indicate whether the goal is satisfied for the topic.

If the team has not reached a consensus by the end of the exploratory interviews, enter "**?**" to indicate that the topic must be investigated in the consolidation plan.

Enter the consensus based on exploratory interviews here

Enter the consensus based on document reviews here.

Carnegie Mellon University
Software Engineering Institute

SCE Validation Worksheet

Projects: A. _Able_   B. _Baker_   C. _Charlie_   D._____

Software Project Planning

Develop Estimates

		Project	Explore Interview	Doc Review	Consolid Interview	Organization
Comments: look at both cost and size estimates		A				
		B				
		C				
		D				
organizational policies		A				
		B				
		C				
		D				
organizational structures		A				
		B				
		C				
		D				

3 If the team reaches a consensus during consolidation interviews or final document reviews, enter the result here.

4 Enter the consensus about the topic for the organization as a whole in this column.

Figure 9-1: Completing a Validation Worksheet

Section 5   Findings

Findings are statements that summarize the teams conclusions about the subprocess areas investigated.

Throughout the site visit, try to articulate conclusions about the information gathered. This helps to test that there is general agreement on what is being heard and to focus the investigation on the objective evidence needed to support the conclusions. These conclusions are referred to as preliminary findings and candidate findings.

A preliminary finding is a statement about the organization's software process capability for which the team has reached a consensus and which is supported by objective evidence. Once a preliminary finding has been made, the topic and or subprocess area is dropped from further consideration. New evidence should still be considered, but you should not spend any more time looking for data relative to the issue.

A candidate finding is one for which there is not yet enough objective evidence to make a decision. Candidate findings become the subjects of consolidation interviews or subsequent document reviews, as shown in Figure 2-1 on page 6-23. If a finding cannot be validated, if doubt remains, or if consensus is not achieved despite additional documentation or interviews, then there can be no finding in that instance and the candidate findings should be discarded.

If you identify a possible weakness, give the development organization an opportunity to produce evidence that might mitigate or eliminate the weakness either by checking with the site visit coordinator (➡page 6-6) or in subsequent interviews. By double checking, the team avoids making findings based on anomalous responses. Do not make a direct mention of the preliminary finding. The request for clarification should define the subject matter
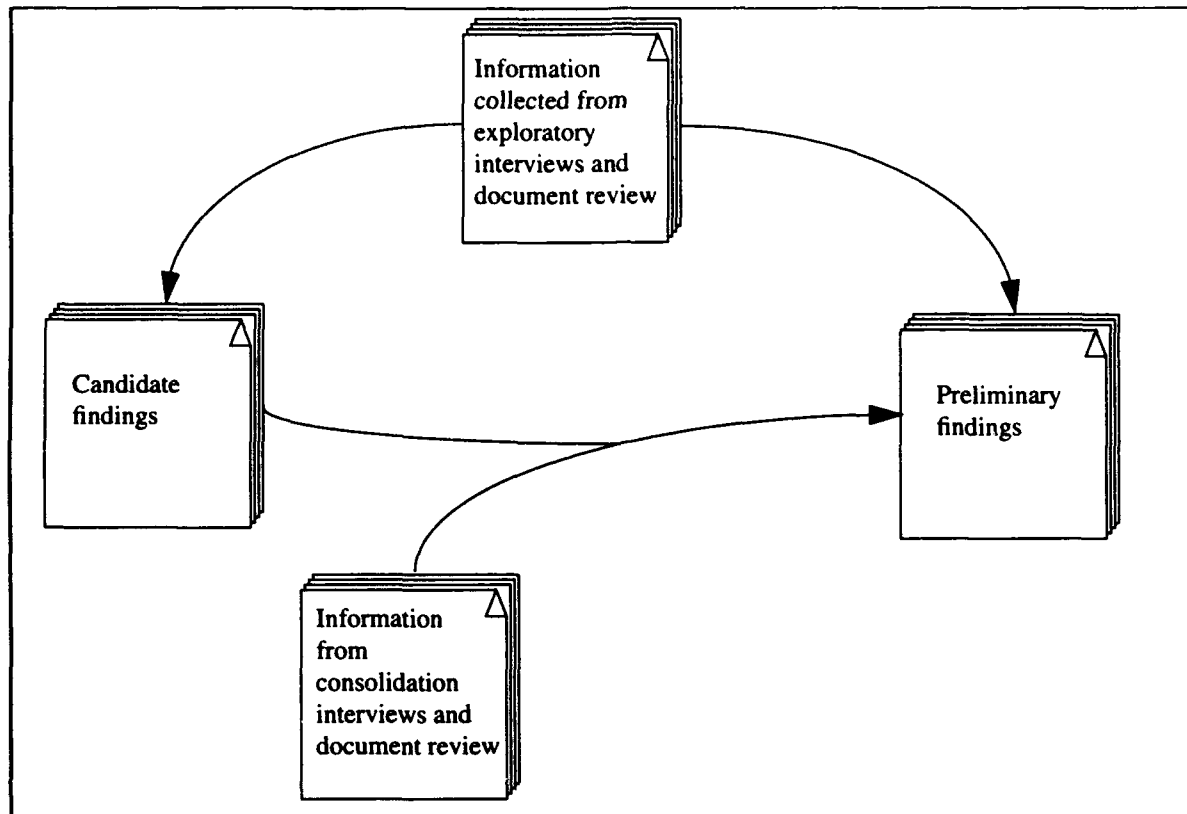
Figure 9-2: Transformation of Information into Findings

and ask whether what the team observed or heard is representative. For example, you might ask "We were not able to determine if the estimates for project size were based on actual data. Did we miss something?"

The final findings are the result of the site visit. They are statements that represent the team's consensus about the organizations software process capability and they are substantiated with objective evidence.

Guidelines   **Determining Findings**

The inputs used to determine the findings are the completed Validation Worksheets. Supporting information may come from Interview Worksheets, and documentation review notes.

Information about planned improvement may come from interviews with organization level managers, project managers, and process group leaders, and from improvement plans. However, verbal plans about improvement are not sufficient. Objective evidence backing up the reality of the plans is necessary to consider whether what the team has heard qualifies as data to be collected.

A finding can be generated for a KPA if the team has reached a consensus for each topic area (subprocess area feature) selected for investigation for the critical subprocess areas of the KPA. The following guidelines should be used when determining findings:

☞ In order for an SCE finding to exist, the following criteria must be met:

- The team must observe supporting evidence in two or more independent sources.

- The team must generate the findings through a consensus process. That is, there are no minority opinions opposed to the finding.

- The objective evidence must support the findings.

☞ All judgements made by the team should be correlated by at least two separate pieces of information.

☞ As the significance of the judgement increases, the correlation may require three or more separate sources of information.

☞ As a general rule, if there is any doubt at all about whether a finding is valid, the organization should be given the benefit of the doubt.

☞ Each KPA in the Target Process Capability must have a finding or an explicit annotation of "no finding," meaning that there was not enough observed evidence to make a finding.

☞ It is possible for a given subprocess area to have strengths, weaknesses and improvement activities.

For example, an organization may have well-defined procedures (a strength), no training in the procedures (a weakness), and an ongoing course development effort for the new procedures sponsored by the organization (an improvement activity).

☞ For each KPA, each topic of investigation for a critical subprocess area is a potential strength, weakness, or both.

☞ For KPAs at the defined level or higher, consider whether the KPAs for the lower levels have been adequately satisfied so that there will be appropriate support.

If they are not adequately satisfied, then the findings should include a statement of this weakness along with any strengths observed for the higher level KPAs.

Guidelines   **How to Express Findings**

☞ Findings are presented in the following format:

- There is one sheet per KPA in the Target Process Capability

- The heading is the KPA name.

- Findings statements in each of the following categories are provided:

  - Summary Statement
  - Strengths
  - Weaknesses
  - Planned Improvements.

☞ Findings should be expressed as observations rather than judgements about the organization's capability.

For example it is better to say:

"Management oversight was observed to depend on the individual styles of each manager. No evidence was found of a corporate approach to management oversight."

rather than:

"The corporate approach to management oversight is inadequate."

☞ Remember you are evaluating a subset of the total projects ongoing at a site as a proxy for predicting the organization's capability to do a specific project, and exceptions may exist because of this process.

☞ Findings should be specific to the point where they identify the cause for a strength or weakness, but not so specific that the finding places the team in a corner by failing to consider exceptions that may exist within the organization.

☞ Be prepared to substantiate the strengths and weaknesses without attributing the information to specific individuals or projects. Individual confidentiality is a vital component of a good site visit.

Chapter 9   Team Caucus
Section 5   Findings
Guidelines   How to Express Findings

Part 3   Appendices

Appendix A   Steps of the SCE Method

This appendix is provided as a quick reference to the phases and steps of the SCE method. It provides information to show at a glance what the purpose of each step is and the relationships between the steps.

The following information is provided:

Name	Page number
Table A-1: Summary of Phases and Steps in an SCE	page A-2
Figure A-1: Overview of Phases in SCE	page A-4
Figure A-2: Diagram of Steps in Phase 1, Evaluation Start	page A-5
Figure A-3: Diagram of Steps in Phase 2. General Preparation	page A-6
Figure A-4: Diagram of Steps in Phase 3, Specific Preparation	page A-7
Figure A-5: Diagram of Steps in Phase 4, Site Data Collection	page A-8
Figure A-6: Diagram of Steps in Phase 5, Findings	page A-9

Table A-1 is a quick reference to the phases and steps of the SCE method. It provides a list of the phases, the steps within the phases, the primary purpose for each step, and a page number for easy reference.

Figures A-1 through A-6 show the relationships among the phases and steps of the SCE method and the flow of information between the steps.

Figure A-1 shows the relationship of the five phases of the SCE method. This diagram provides the context for the other diagrams

Figures A-2 through A-6 show the relationships of the steps within each of the phases.

Phase	Step	Purpose	Page
Phase 1: Evaluation Start	1. Develop Target Product Profile	Understand attributes of the software product and the project required to produce it.	page 4-15
	2. Determine Target Process Capability	Determine the process capability that is most appropriate for the planned development—the Target Process Capability.	page 4-16
	3. Select Team	Have a trained team in place to execute the SCE.	page 4-18
Phase 2: General Preparation	4. Create Experience Table	Identify areas where the development organizations lack experience, indicating a potential for risk.	page 5-3
	5. Create Critical Subprocess Area List	Define and document the scope of the SCE, in terms of critical subprocess areas within the Target Process Capability KPAs.	page 5-13
	6. Originate Validation Worksheets	Record the set of critical subprocess areas for all development organizations on forms that can be used in subsequent information collection efforts.	page 5-26
Phase 3: Specific Preparation	7. Select Projects to Investigate	Select projects for evaluation that give the most insight into the processes that will be used.	page 5-31
	8. Develop Key Issue Worksheet	Create a consolidated list of key issues for investigation at the development organization site.	page 5-34
	9. Develop Topic Lists	Select topics for probing the process implementation; topics define observable work practices that map to the critical subprocess areas.	page 5-39
	10. Add Topics to Validation Worksheet	Capture the consolidated topic list for use at a particular site.	page 5-41
	11. Prepare for Exploratory Interviews	Develop detailed interview strategy, including the team's decisions on who will be interviewed, when they will be interviewed, and what they will be asked.	page 5-44
	12. Prepare Entry Briefing	Establish the agenda for the initial organization meeting and set initial expectations for the site visit.	page 5-52

Table A-1: Summary of Phases and Steps in an SCE

Phase	Step	Purpose	Page
Phase 4: Site Data Collection	13. Conduct Initial Organization Meeting	Clarify expectations of the SCE site visit.	page 6-13
	14. Conduct Initial Document Review	Determine the degree to which the organization and project-level documentation define and support standard processes for the KPAs and subprocess areas under investigation.	page 6-15
	15. Conduct Exploratory Interviews	Provide insight into how the subprocess areas are implemented in practice; determine the extent that processes have been internalized by the development organizations; identify critical implementation-level documents.	page 6-18
	16. Hold Team Caucus	Analyze, share, and consolidate information in order to reach conclusions about topics.	page 6-20
	17. Conduct Document Review	Search for objective evidence of how processes are implemented at the working level.	page 6-24
	18. Develop Preliminary Findings	Articulate conclusions about the subprocess areas based on the information available; guide subsequent information-gathering efforts.	page 6-24
	19. Create Consolidation Plan	Plan and initiate further data collection.	page 6-27
	20. Conduct Consolidation Interviews	Clarify any remaining issues by confirming or negating candidate findings through further interviews.	page 6-30
	21. Conduct Final Document Review	Clarify any remaining issues by confirming or negating candidate findings through further document review.	page 6-31
Phase 5: Findings	22. Determine Findings	Validate the preliminary findings and consolidate them by KPA.	page 6-34
	23. Produce Findings Report	Document the SCE activities and provide a formal record of the findings.	page 6-36
	24. Conduct Exit Briefing	Provide feedback to the recipient and conclude the SCE.	page 6-38

Table A-1: Summary of Phases and Steps in an SCE (Continued)

Figure A-1: Overview of Phases in SCE

This diagram shows the information flow between the 5 phases

This diagram shows the information flow between steps, not the sequence of activity.

Note: The inputs to Step 1 are omitted from this diagram, as is the output of Step 3 (the SCE team).

Phase 1 Evaluation Start

Maturity Model

Step 1
Develop
Product Profile

Target Product Profile

Maturity levels,
Key process areas

Step 2
Determine Target
Process Capability

Target Process Capability

Step 3
Select
Team

Phase 3
Specific
Preparation

Phase 2
General
Preparation

Figure A-2: Diagram of Steps in Phase 1, Evaluation Start

Figure A-3:   Diagram of Steps in Phase 2. General Preparation

Figure A-4: Diagram of Steps in Phase 3, Specific Preparation

This diagram shows the information flow between steps, not the sequence of activity.

Note: The term "previous information" is used to refer to information collected during the site visit, including completed Interview Worksheets and document review working notes. Refer to the step descriptions for detail.

Phase 4 Site Data Collection

Figure A-5:   Diagram of Steps in Phase 4, Site Data Collection

This diagram shows the information flow between steps, not the sequence of activity.

Phase 5 Findings

Phase 4
Site Data Collection

Findings Report

Briefing
• Strengths
• Weaknesses

Preliminary findings,
Completed Validation Worksheets,
Completed Interview Worksheets,
Document review working notes

Completed Validation Worksheets,
Completed Interview Worksheets,
Document review working notes

Step 22
Determine Findings

Step 23
Produce Findings Report

Step 24
Conduct Exit Briefing

Final Findings

CMM KPA Goals
Probing Guides

SCE Method

Figure A-6: Diagram of Steps in Phase 5, Findings

Appendix B   Maturity Model

Contents

Version 2.0 of the SCE Method uses the process maturity model defined in the *Capability Maturity Model for Software V1.1* (CMM) [Paulk 93a]. This appendix provides a summary of essential information contained in the CMM or derived from the CMM which is used in the SCE Method. The CMM information is repeated in this document for easy reference. This appendix contains the following sections:

Section	Page number
B.1 - CMM V1.1 Process Maturity Levels	page B-2
B.2 - CMM V1.1 Key Process Areas (KPAs)	page B-3
B.3 - CMM V1.1 KPA Goals	page B-4
B.4 - Subprocess Areas	page B-10
B.5 - Features	page B-16

The first three sections summarize the levels, KPAs, and KPA goals from the CMM, Version 1.1. The last two sections contain subprocess areas and feat _ ·: Both of these sections are extracted from the commu _ rating framework for CMM-based appraisals which is under development at the SEI.

The subprocess areas used in SCE are derived from the goals of the CMM; each subprocess area corresponds to a single goal, and each goal has a single subprocess area. The features include the common features from the CMM and additional subfeatures.

Section B.1   CMM V1.1 Process Maturity Levels

A maturity level is "a well-defined evolutionary plateau toward achieving a mature software process" [Paulk 93b]. The SCE Method uses these definitions of process maturity levels, which are extracted from *Capability Maturity Model for Software, Version 1.1* [Paulk 93a]:

Process Maturity Levels	Descriptions
1-Initial	The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort.
2-Repeatable	Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.
3-Defined	The software process for both management and engineering activities is documented, standardized, and integrated into a standard software process for the organization. All projects use an approved, tailored version of the organization's standard software process for developing and maintaining software.
4-Managed	Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.
5-Optimized	Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.

Table B-1: CMM V1.1 Process Maturity Levels

Section B.2 CMM V1.1 Key Process Areas (KPAs)

A key process area (KPA) "identifies a cluster of related activities that, when performed collectively, achieve a set of goals considered important for enhancing process capability" [Paulk 93b]. The KPAs used in Version 2.0 of the SCE Method are from *Capability Maturity Model for Software, Version 1.1* [Paulk 93a].

Process Maturity Levels	KPAs
5 - Optimized	Defect Prevention
	Technology Change Management
	Process Change Management
4 - Managed	Quantitative Process Management
	Software Quality Management
3 - Defined	Organization Process Focus
	Organization Process Definition
	Training Program
	Integrated Software Management
	Software Product Engineering
	Intergroup Coordination
	Peer Reviews
2 - Repeatable	Requirements Management
	Software Project Planning
	Software Project Tracking and Oversight
	Software Subcontract Management
	Software Quality Assurance
	Software Configuration Management
1 - Initial	

Table B-2: CMM V1.1 KPAs

Section B.3   CMM V1.1 KPA Goals

The goals of a KPA are "a summary of the key practices of a KPA and can be used to det   ether an organization or project has effectively implemented the KPA. The goals signify the scope, boundaries, and intent of each KPA" [Paulk 93b]. The KPA goals used in Version 2.0 of the SCE Method are from *Capability Maturity Model for Software, Version 1.1* [Paulk 93a].

Goals for the Repeatable Level KPAs
Requirements Management
Goal 1 System requirements allocated to software are controlled to establish a baseline for software engineering and management use.
Goal 2 Software plans, products, and activities are kept consistent with the system requirements allocated to software.
Software Project Planning
Goal 1 Software estimates are documented for use in planning and tracking the software project.
Goal 2 Software project activities and commitments are planned and documented.
Goal 3 Affected groups and individuals agree to their commitments related to the software project.
Software Project Tracking and Oversight
Goal 1 Actual results and performances are tracked against the software plans.
Goal 2 Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans.
Goal 3 Changes to software commitments are agreed to by the affected groups and individuals.
Software Subcontract Management
Goal 1 The prime contractor selects qualified software subcontractors.
Goal 2 The prime contractor and the software subcontractor agree to their commitments to each other.
Goal 3 The prime contractor and the software subcontractor maintain ongoing communications.
Goal 4 The prime contractor tracks the software subcontractor's actual results and performance against its commitments.

Table B-3: Goals for the Repeatable Level KPAs

Goals for the Repeatable Level KPAs (Continued)
Software Quality Assurance
Goal 1   Software quality assurance activities are planned.
Goal 2   Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.
Goal 3   Affected groups and individuals are informed of software quality assurance activities and results.
Goal 4   Noncompliance issues that cannot be resolved within the software project are addressed by senior management.
Software Configuration Management
Goal 1   Software configuration management activities are planned.
Goal 2   Selected software work products are identified, controlled, and available.
Goal 3   Changes to identified software work products are controlled.
Goal 4   Affected groups and individuals are informed of the status and content of software baselines.

Table B-3: Goals for the Repeatable Level KPAs (Continued)

Goals for the Defined Level KPAs
Organizational Process Focus   Goal 1   Software process development and improvement activities are coordinated across the organization.   Goal 2   The strengths and weaknesses of the software processes used are identified relative to a process standard.   Goal 3   Organization-level process development and improvement activities are planned.
Organizational Process Definition   Goal 1   A standard software process for the organization is developed and maintained.   Goal 2   Information related to the use of the organization's standard software process by the software projects is collected, reviewed, and made available.
Training Program   Goal 1   Training activities are planned.   Goal 2   Training for developing the skills and knowledge needed to perform software management and technical roles is provided.   Goal 3   Individuals in the software engineering group and software-related groups receive the training necessary to perform their roles.
Integrated Software Management   Goal 1   The project's defined software process is a tailored version of the organization's standard software process.   Goal 2   The project is planned and managed according to the project's defined software process.
Software Product Engineering   Goal 1   The software engineering tasks are defined, integrated, and consistently performed to produce the software.   Goal 2   Software work products are kept consistent w⁻   ⁻h other.

Table B-4: Goals for the Defined Level KPAs

Goals for the Defined Level KPAs (Continued)
Intergroup Coordination Goal 1   The customer's requirements are agreed to by all affected groups. Goal 2   The commitments between the engineering groups are agreed to by the affected groups. Goal 3   The engineering groups identify, track, and resolve intergroup issues.
Peer Reviews Goal 1   Peer review activities are planned. Goal 2   Defects in the software work products are identified and removed.

Table B-4: Goals for the Defined Level KPAs (Continued)

Goals for the Managed Level KPAs
Quantitative Process Management Goal 1   The quantitative process management activities are planned. Goal 2   The process performance of the project's defined software process is controlled quantitatively. Goal 3   The process capability of the organization's standard software process is known in quantitative terms.
Software Quality Management Goal 1   The project's software quality management activities are planned. Goal 2   Measurable goals for software product quality and their priorities are defined. Goal 3   Actual progress toward achieving the quality goals for the software products is quantified and managed.

Table B-5: Goals for the Managed Level KPAs

Goals for the Optimized Level KPAs
Defect Prevention
Goal 1 Defect prevention activities are planned.
Goal 2 Common causes of defects are sought out and identified.
Goal 3 Common causes of defects are prioritized and systematically eliminated.
Technology Change Management
Goal 1 Incorporation of technology changes are planned.
Goal 2 New technologies are evaluated to determine their effect on quality and productivity.
Goal 3 Appropriate new technologies are transferred into normal practice across the organization.
Process Change Management
Goal 1 Continuous process improvement is planned.
Goal 2 Participation in the organization's software process improvement activities is organization wide.
Goal 3 The organization's standard software process and the projects' defined software processes are improved continuously.

Table B-6: Goals for the Optimized Level KPAs

Section B.4 Subprocess Areas

The KPAs defined in the CMM are large clusters of activities with multiple goals. In order to understand the processes implemented by an organization and to make judgments about them, it is convenient to divide the KPAs into smaller chunks of activities. The SCE Method uses subprocess areas for this purpose. The subprocess areas listed here were developed as part of the common rating framework development at the SEI.

A subprocess area is a set of activities in an implemented process that, acting together, works to achieve one of the goals of a KPA. There is a one-to-one correspondence between the subprocess areas and the KPA goals and the subprocess area definitions are derived from the goal statement.

The following tables list the KPAs, subprocess areas, actions taken in accordance with the subprocess areas, and the KPA goals which correspond to the subprocess areas. The tables are organized by maturity level.

Appendix F, Subprocess Area Selection Tables, defines the relationship between the subprocess areas listed below and the attributes in the profiles used in SCE (such as the Proposed Product Profile and the Project Profiles from projects that are candidates for evaluation).

Subprocess Areas for the Repeatable Level KPAs

KPA	Subprocess Area Name	Subprocess Area Action	Corresponding KPA Goal
Requirements Management	**Establish and maintain requirements baseline**	Establish and maintain a baseline of agreed-upon requirements allocated to software.	1. System requirements allocated to software are controlled to establish a baseline for software engineering and management use.
	Manage requirements-driven changes	Manage requirements-driven changes.	2. Software plans, products, and activities are kept consistent with the system requirements allocated to software.
Software Project Planning	**Develop estimates**	Develop documented estimates.	1. Software estimates are documented for use in planning and tracking the software project.
	Plan software activities	Plan and document software activities and commitments.	2. Software project activities and commitments are planned and documented.
	Make commitments	Obtain agreement on planned commitments.	3. Affected groups and individuals agree to their commitments related to the software project.
Software Project Tracking and Oversight	**Track progress**	Track progress against software plans.	1. Actual results and performances are tracked against the software plans.
	Take corrective action	Take and manage corrective actions to reduce variance from plans.	2. Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans.
	Manage commitment changes	Obtain agreement on commitment changes.	3. Changes to software commitments are agreed to by the affected groups and individuals.

Table B-7: Subprocess Areas for the Repeatable Level KPAs

Subprocess Areas for the Repeatable Level KPAs (Continued)

KPA	Subprocess Area Name	Subprocess Area Action	Corresponding KPA Goal
Software Subcontract Management	Select subcontractors	Select qualified subcontractors.	1. The prime contractor selects qualified software subcontractors.
	Establish and maintain commitments	Obtain and maintain agreement by contractor and subcontractor for mutual commitments.	2. The prime contractor and the software subcontractor agree to their commitments to each other.
	Maintain communications	Maintain communications with subcontractor.	3. The prime contractor and the software subcontractor maintain ongoing communications.
	Track progress	Track subcontractor progress.	4. The prime contractor tracks the software subcontractor's actual results and performance against its commitments.
Software Quality Assurance	Plan SQA	Plan software quality assurance activities.	1. Software quality assurance activities are planned.
	Perform SQA	Verify adherence of activities and products to applicable standards.	2. Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.
	Communicate results	Communicate SQA results.	3. Affected groups and individuals are informed of software quality assurance activities and results.
	Address noncompliance	Address noncompliance issues.	4. Noncompliance issues that cannot be resolved within the software project are addressed by senior management.
Software Configuration Management	Plan SCM	Plan software configuration management activities.	1. Software configuration management activities are planned.
	Create software work products baselines	Identify selected software work products for a baseline, which is controlled and made available.	2. Selected software work products are identified, controlled, and available.
	Control changes	Control changes to software baselines.	3. Changes to identified software work products are controlled.
	Report status	Report software configuration status.	4. Affected groups and individuals are informed of the status and content of software baselines.

Table B-7: Subprocess Areas for the Repeatable Level KPAs (Continued)

Subprocess Areas for the Defined Level KPAs

KPA	Subprocess Area Name	Subprocess Area Action	Corresponding KPA Goal
Organization Process Focus	**Coordinate software process activities**	Coordinate software process development and improvement activities.	1. Software process development and improvement activities are coordinated across the organization.
	Assess software processes used	Assess software processes in use against a process standard.	2. The strengths and weaknesses of the software processes used are identified relative to a process standard.
	Plan SPI	Plan software process development and improvement activities.	3. Organization-level process development and improvement activities are planned.
Organization Process Definition	**Provide standard process**	Develop and maintain the organization's standard software process.	1. A standard software process for the organization is developed and maintained.
	Retain software process information	Collect, review, and make available the organizational software process data.	2. Information related to the use of the organization's standard software process by the software projects is collected, reviewed, and made available.
Training Program	**Plan training**	Plan training activities.	1. Training activities are planned.
	Provide training.	Provide training.	2. Training for developing the skills and knowledge needed to perform software management and technical roles is provided.
	Receive necessary training.	Receive necessary training.	3. Individuals in the software engineering group and software-related groups receive the training necessary to perform their roles.
Integrated Software Management	**Define project process**	Define project's software process by tailoring the organization's standard software process.	1. The project's defined software process is a tailored version of the organization's standard software process.
	Manage according to process	Manage project according to its defined process.	2. The project is planned and managed according to the project's defined software process.
Software Product Engineering	**Build software**	Build and maintain software according to project's defined software process.	1. The software engineering tasks are defined, integrated, and consistently performed to produce the software.
	Ensure consistency	Ensure consistency of software work products.	2. Software work products are kept consistent with each other.

Table B-8: Subprocess Areas for the Defined Level KPAs

Subprocess Areas for the Defined Level KPAs (Continued)

KPA	Subprocess Area Name	Subprocess Area Action	Corresponding KPA Goal
Intergroup Coordination	Agree on customer's requirements	Obtain agreement on customer's requirements.	1. The customer's requirements are agreed to by all affected groups.
	Coordinate intergroup commitments	Obtain agreement by affected groups on commitments between engineering groups.	2. The commitments between the engineering groups are agreed to by the affected groups.
	Manage intergroup issues	Identify, track, and resolve intergroup issues.	3. The engineering groups identify, track, and resolve intergroup issues.
Peer Reviews	Plan peer reviews	Plan peer review activities	1. Peer review activities are planned.
	Identify and remove defects	Identify and remove defects in the software work products	2. Defects in the software work products are identified and removed.

Table B-8: Subprocess Areas for the Defined Level KPAs (Continued)

Subprocess Areas for the Managed Level KPAs

KPA	Subprocess Area Name	Subprocess Area Action	Corresponding KPA Goal
Quantitative Process Management	Plan QPM	Plan quantitative process management for the project.	1. The quantitative process management activities are planned.
	Control process quantitatively	Control project's process quantitatively.	2. The process performance of the project's defined software process is controlled quantitatively.
	Establish organization's process capability	Analyze and combine process performance of an organization's projects.	3. The process capability of the organization's standard software process is known in quantitative terms.
Software Quality Management	Plan quality management	Plan software quality management.	1. The project's software quality management activities are planned.
	Define software quality goals	Define measurable, prioritized quality goals.	2. Measurable goals for software product quality and their priorities are defined.
	Track quality progress	Track progress toward achieving quality goals.	3. Actual progress toward achieving the quality goals for the software products is quantified and managed.

Table B-9: Subprocess Areas for the Managed Level KPAs

Subprocess Areas for the Optimized Level KPAs

KPA	Subprocess Area Name	Subprocess Area Action	Corresponding KPA Goals
Defect Prevention	Plan defect prevention	Plan defect prevention activities.	1. Defect prevention activities are planned.
	Identify defect causes	Identify common causes of defects.	2. Common causes of defects are sought out and identified.
	Eliminate defect causes	Prioritize and eliminate causes of defects.	3. Common causes of defects are prioritized and systematically eliminated.
Technology Change Management	Plan technology changes	Plan incorporation of technology changes.	1. Incorporation of technology changes are planned.
	Evaluate new technologies	Determine effect of new technologies on quality and productivity.	2. New technologies are evaluated to determine their effect on quality and productivity.
	Adopt new technology	Transfer appropriate new technologies into practice.	3. Appropriate new technologies are transferred into normal practice across the organization.
Process Change Management	Plan process improvement	Plan continuous process improvement.	1. Continuous process improvement is planned.
	Empower everyone	Empower organization people to participate in process improvement.	2. Participation in the organization's software process improvement activities is organization wide.
	Continuously improve	Continuously identify and manage process improvement implementations.	3. The organization's standard software process and the projects' defined software processes are improved continuously.

Table B-10: Subprocess Areas for the Optimized Level KPAs

Section B.5   Features

A subprocess area is inherently too broad to investigate within the constraints of a site visit. However, each subprocess area has common features. Common features are "attributes that indicate whether the implementation and institutionalization of a key process is effective, repeatable and lasting." In other words, a common feature is an implementation characteristic common to all subprocess areas.

The features used in the SCE Method are based on the definitions of the common features from the *Capability Maturity Model for Software, Version 1.1* [Paulk 93a]. As part of the common rating framework development at the SEI, the common features were decomposed into the 13 features described in subparagraphs (a) through (m) below.

1. **Commitment to Perform** (the actions taken to ensure that the subprocess area is implemented and will endure)

    a. **leadership** - the assignment of responsibility and the presence of sponsorship

    b. **organizational policies** - there are written policies governing the subprocess area

2. **Ability to Perform** (the preconditions to implement the subprocess area competently exist in the project or organization)

    c. **resources** - the adequacy of resources (e.g., staff, funds, facilities, tools)

    d. **organizational structures** - the organizational structure provides support for the process activities (e.g., job descriptions, defined relationships between entities on the organization chart)

e. **training** - availability of pertinent training and orientation, and its timeliness for the people who carry out the activities in the implementation of the subprocess area (e.g., curriculum content, training schedule, records)

3. **Activities Performed** (the roles and procedures necessary for implementation of the processes)

   f. **plans and procedures** - plans and procedures exist and are prepared according to a documented procedure

   g. **work performed** - the objective evidence of the use of plans, procedures, and standards in the work done by the organization (i.e., the track record and "paper or electronic trail")

   h. **tracking** - how the work is tracked and how problems are identified

   i. **corrective actions** - the identification and resolution of problems

4. **Measurement and Analysis** (the determination of the status and effectiveness of the activities)

   j. **measure process** - the measurements of activities performed (e.g., resources consumed, problems encountered, work product characteristics, and status of activities)

   k. **analyze measurements** - the analysis and use of measurements taken

5. **Verifying Implementation** (the actions that ensure compliance to established practice)

   l. **reviews** - management reviews

   m. **audits** - there are audits undertaken of activities and work products

Features provide a level of structure that enables teams to ask specifically focused yet open-ended questions during interviews and document reviews.

When a feature is tied to a specific subprocess area it becomes a topic for investigation. A *topic* is an abstraction of a work practice. Topics are intended to be detailed enough to focus the investigation on observable, documented work practices, but sufficiently abstract that they avoid prescribing how the topic is implemented.

The features are used in Step 9, "Develop Topic Lists" (➥page 5-39), along with the "Look For" tables (➥Appendix G), to specify the topics which will be investigated for each subprocess area on the Critical Subprocess Area List.

Appendix C   Attribute Definitions

This appendix contains the definitions of the standard product and project attributes as they are used during the first three phases of the SCE method (Evaluation Start, General Preparation, and Specific Preparation). The attributes are used to specify important characteristics of a product or project so that comparisons can be made in a systematic way.

Major Attributes

The major attributes are used to compare previous experience on the part of the development organization and end user to the experience needed for the current development. This comparison is used to identify potential risk areas that should be looked at during the SCE. The major attributes are also given first consideration when selecting projects for evaluation.

The major attributes are used in creating the Target Product Profile, the Proposed Project Profile, the Project Profiles, the Mismatch Identification Table, the Experience Table, the Key Issue Table, and the Key Issue Worksheet. They are also used as a guide for selecting subprocess areas from the Subprocess Area Selection Tables shown in Appendix F.

Application domain

The *application domain* attribute indicates the area of subject matter expertise needed to translate system requirements into software requirements.

There is no accepted taxonomy of application domains; however, the concept is widely understood and used. Information systems, command and control systems, weapon systems, simulation systems, training systems, avionic systems, sensing systems, and so on are all recognized and accepted as different application domains. What makes application domains different is the operational environment that uses the system. The unique characteristics of the operational environment are

- The mission for which the system is needed.

- The roles and responsibilities of the people who interface with the system.

- The resources that the system depends upon, which defines the potential limit of the services that the system can provide the people in the operational environment.

Product type

The *product type* attribute refers to the particular aspect of the application domain which the system will support or to the type of service which the system will provide. It may be considered a subset of the application domain.

For example, communications or displays could be product types in a command and control system, a weapons system or other application domain. Although there may be similarities in the communications subsystem in the various application domains, they each have their own set of unique problems which must be addressed.

Size

The size attribute is composed of three related attributes. The *contract duration* is the estimated or required length of time for the development of the software product. The *software team size* is the number of software developers who will be involved in the project. The *estimated software size* is the amount of code to be developed.

There is no standard way of measuring the size attributes. For the purposes of an SCE, the specific method used is not important as long as the method is used consistently so that comparisons will be meaningful.

This attribute was previously referred to as "Product Size," and abbreviated "Ps"; in some of the materials the abbreviation "Ps" is still used.

Type of work

The *type of work* attribute is used to indicate the portion of the development life cycle which will be performed by the development organization. The life cycle can be an important consideration. For example, consider a maintenance shop planning a new software development that starts with requirements analysis and design. Because the development organization is proposing development activities for a portion of the life cycle that the organization does not have extensive experience with, there may be increased risk for the planned development.

The type of work attribute may indicate subprocess areas that should receive more or less emphasis during an SCE. Similar factors might apply if an organization was going to use a new life cycle model (or development methodology) for a planned development.

The following are examples of different types of work that may be required:

- Full Software Development: The development organization is required to build a product based upon the system requirements. The development organization will typically be required to complete software requirements, top level design, detailed design, code and unit test, and acceptance testing at the development organization's site. The development scope is the same as or similar to the phases described in DoD-STD-2167A.

- Code Development Only: The development organization is required to develop code according to the system requirements and software top level design provided by the issuing authority. This type of development might be done under a delivery order contract. The development organization may do the detailed design, coding, integration, and testing, but the system testing may be done by the customer.

- System Development Without Coding: The development organization may be required to do all the work except the software detailed design and development.

- A Prime Contract Acquisition: In a large system acquisition there may be many organizations who subcontract significant parts of the system, especially software parts. The prime contractor allocates system requirements to the subcontractor, integrates the components, and conducts acceptance tests.

Subcontractors

The *subcontractors* attribute is used to indicate whether the development organization plans to use subcontractors. If the development organization intends to use subcontractors for the planned development and does not have demonstrated experience using subcontractors, then this attribute is a factor. The lack of experience indicates that there may be risk in areas such as requirements management and software configuration management because of the additional coordination of effort required. If there are no plans to use subcontractors, then the lack of experience in subcontract management does not need to be considered.

The subcontractors attribute does not replace the Software Subcontract Management KPA of the CMM. The Software Subcontract Management KPA applies anytime the development organization plans to use subcontractors for a major, separately managed portion of the software development, regardless of the development organization's experience with handling subcontractors. If the

development organization lacks experience, then the subcontractors attribute is used to indicate an even greater potential risk that applies to other KPAs and subprocess areas as well.

Operational precedence

The *operational precedence* attribute indicates whether the end user has previous experience with the type of system to be built. The values for this attribute are *no* (meaning operational precedence is not a factor—the end user has experience with similar systems), or *yes* (meaning the system is unprecedented to the end user.) Systems that are providing a new capability tend to have more changes to the requirements than systems that are replacing existing systems. The more unprecedented a system is, the more dynamic the requirements will be.

Minor Attributes

The minor attributes are used on the Target Product Profile, the Proposed Project Profile, the Project Profiles, the Mismatch Identification Table, and the Experience Table. They provide additional information which may be used in selecting projects for evaluation.

Language(s)

The *language* attribute indicates the programming languages in which the code is to be written, or in which it has been written.

Target

The *target* attribute indicates the hardware configuration that the developed software will run on when operational.

Applicable standards   The *applicable standards* attribute indicates the development standards that are imposed on the project such as DoD-STD-2167A, DoD-STD-2168, or MIL-STD-1521B.

Customer   The *customer* attribute indicates who the development is being done for. Examples include one of the DoD services or a particular market within industry.

Host development system   The *host system* attribute refers to the computer environment which will be used for the software development.

Configuration management tool   The *configuration management tool* attribute defines the tool set used on the host development system for supporting such activities as the software build process, baselining, and version control.

Schedule Attributes

The schedule attributes are used on the Project Profiles. They identify where the development organization is in relation to the project's schedule. The schedule attributes are used in selecting projects to be evaluated.

Current Phase   The *current phase* attribute refers to the life cycle phase of the development which the project is currently in, such as design, coding, integration, or acceptance testing.

Current Month   The *current month* attribute is the number of months since the start of the project.

Start

The *start* attribute shows when the project actually begins relative to the start of the contract.

Design Ends

The *design ends* attribute shows how long after the start of the project the design phase was completed or is scheduled to be completed.

Coding Ends

The *coding ends* attribute shows how long after the start of the project the coding phase was completed or is expected to be completed.

Appendix D   Sample Forms

This appendix provides examples of the forms used for planning, analysis, and data collection throughout the SCE process. The forms included here are based on the ones used during the SCE team training; in some cases they have been resized to fit in this document better.[1] These forms are conceptual in nature; they indicate information needed to conduct an SCE, but they are not mandatory.

Examples of the following forms are shown in this appendix:

A sample copy of each form is included along with the purpose of the form, a summary of how the form is used, and a description of the data recorded on the form.

---

1.   The terminology of the forms is acquisition oriented because that was the focus of the initial training, and is still the primary use of the SCE method. For example, "offeror" is used for "development organization" on some of the forms.

Section D.1   Target Product Profile

The Target Product Profile is used to specify the characteristics of the product to be developed in terms of a standard set of attributes (the attributes are defined in Appendix C). The Target Product Profile represents a "customer view" of the product to be built. The Target Product Profile is used to identify risk areas that should be given special attention during the evaluation, to define expertise needed on the SCE team, and to provide a the team with a basic understanding of the desired product. Figure D-1 shows a Target Product Profile form with sample data.

Target Product Profile

Attributes	RFP Development
Major Attributes	
Application Domain	Command and Control
Product Type	ASW helicopters/sonobuoys

Size
   Contract Duration
   Software Team Size
   Estimated Software Size (KSLOC)
	
	24 months
	100
	300

Type of Work	full development
Operational Precedence	no – replacement of existing system
Minor Attributes	
Language(s)	Ada
Target	M68000
Applicable Standards	DoD-STD-2167A, 2168
Customer	Navy

Figure D-1: Sample Target Product Profile Form

The Target Product Profile is developed in Step 1 (➡page 4-15) at the start of the SCE process. It is created by the sponsoring organization. The data for the form is based on the sponsoring organization's independent cost and schedule estimates. In source selection, most of the Target Product Profile information is contained in the Request For Proposal (RFP). One Target Product Profile is developed during an SCE.

The Target Product Profile is used in Step 2 (➡page 4-16) to determine the Target Process Capability. It is used in Step 3 (➡page 4-18) to show the types of experience and background to look for when selecting team members. The *operational precedence* attribute from the form is also used in Step 5 (➡page 5-13) for selecting critical subprocess areas. In Step 5, the Target Product Profile is also used to compare the sponsoring organization's view of the product to be built with the development organization's view. The Target Product Profile may also be used as an additional input in Step 4 (➡page 5-3) for creating the Experience Table and in Step 7 (➡page 5-31) for selecting projects for evaluation.

A Target Product Profile lists the names of the attributes and the characteristics of the product in terms of the attributes. The Target Product Profile uses all the major attributes except *subcontractors* and all the minor attributes except *host development system* and *configuration management tool*.[1]

---

1.   The *host development system* and *configuration management tool* attributes are normally not specified by the sponsoring organization, and may be different for each development organization.

Section D.2   Proposed Project Profile

The Proposed Project Profile is developed by a development organization to describe the planned development. The Proposed Project Profile provides a "developer view" of the planned development. The information is specified in terms of a standard set of attributes (the attributes are defined in Appendix C). The information is used to help evaluate a development organization's previous experience relative to the product being procured in order to identify risk areas that should be given special attention during the evaluation. The information is also used to help select projects for evaluation. Figure D-2 shows a Proposed Project Profile form with sample data.

When the decision to use SCE has been made, the sponsoring organization will request that each of the development organizations prepare a Proposed Project Profile. There will be one Proposed Project Profile for each development organization.

In source selection, the data required for the Proposed Project Profile should be described in the RFP.

The Proposed Project Profile is used in Step 4 (➡page 5-3) along with the Project Profiles to create the Experience Table. The Proposed Project Profile is also used in Step 7 (➡page 5-31) as a guide for selecting projects for evaluation.

A Proposed Project Profile lists the names of the attributes and the characteristics of the project in terms of the attributes. The Proposed Project Profile uses all the major attributes, except for *operational precedence*,[1] and all of the minor attributes.

---

1. *Operational precedence* is an indication of whether the end user has previous experience with the type of system to be built. It does not depend on the experience of the development agency.

Proposed Project Profile

Attributes	Proposed Development
Major Attributes	
Application Domain	Command and Control
Product Type	ASW helicopters/sonobuoys

Size
  Contract Duration
  Software Team Size
  Estimated Software Size (KSLOC)
	
	24 months
	40
	130 (90 new, 40 port/mod)

Type of Work	full development
Subcontractors	none expected
Minor Attributes	
Language(s)	Ada (new), FORTRAN and Assembly (ported)
Target	M68000
Applicable Standards	DoD-STD-2167A, DoD-STD-2168
Customer	Navy
Host Development System	VAX/VMS
Configuration Management Tool	CMS/ MMS

Figure D-2: Sample Proposed Project Profile Form

Section D.3   Project Profiles

The Project Profiles are similar to the Target Product Profile and the Proposed Project Profile, but are derived from information about actual projects rather than estimates about planned efforts. They are used to gather high level project information from a development organization about previous and current projects. The information shows experience that is relevant to the planned development. The Project Profiles are used along with the Proposed Project Profile to compare a development organization's previous experience to the planned development effort in order to identify risk areas that should be given special attention during the evaluation. The information is also used to help select projects for evaluation. Figure D-3 below shows Project Profiles for three projects with sample data.

The sponsoring organization will request that each development organization prepare Project Profiles for six to eight projects which are similar to the proposed project. The Project Profiles are used in Step 4 (➥page 5-3) along with the Proposed Project Profile to create the Experience Table. They are also used in Step 7 (➥page 5-31) as a guide for selecting projects for evaluation and in Step 11 (➥page 5-44) to help generate the detailed interview plan.

The first column of the Project Profile lists the names of the attributes. A Project Profile uses all the major attributes, except for *operational precedence*,[1] all of the minor attributes, and the schedule attributes. (The attributes are defined in Appendix C).

---

1. *Operational precedence* is an indication of whether the end user has previous experience with the type of system to be built. It does not depend on the experience of the development organization.

Project Profiles

Project	Able	Baker	Charlie
Major Attributes			
Application Domain	acoustic signal processing	acoustic signal processing	command and control
Product Type	sonar navigation (upgrade)	sonar signal analysis (upgrade)	helicopter drone (subcontractor to Mega Corp)
Size			
Contract Duration	27 months	27 months	29 months
Software Team Size	37	34	27
Estimated Software Size (KSLOC)	160 (80 new, 80 port/mod)	150 (110 new, 40 port/mod)	125 (all new)
Type of Work	full development	full development	code development
Subcontractors	none	none	none
Minor Attributes			
Language(s)	CMS-2, assembly	Ada, Fortran	Ada
Target	UYK-43	VAX	M68000
Applicable Standards	DoD-STD-1679A	DoD-STD-2167	DoD-STD-2167A
Customer	Navy	Navy	Navy
Host Development System	Univac 1100	VAX/VMS	VAX/VMS
Configuration Management Tool	Sigma Tech Tool	CMS and MMS (VAX tools)	CMS and MMS (VAX tools)
Schedule Data			
Current Phase	system testing	integration and test	coding
Current Month	25	21	18
Start	month 0	month 0	month 0
Design Ends	month 13	month 13	month 15, slipped to month 17
Coding Ends	month 20	month 20	month 22

Figure D-3: Sample Project Profiles Form

Next, the Project Profile contains a column for each project that lists the characteristics of the projects in terms of the attributes.

Section D.4   Mismatch Identification Table

The Mismatch Identification Table is a tool used to analyze the experience of a specific development organization relative to the product being procured. A Mismatch Identification Table is prepared for each specific development organization. Figure D-4 is a sample Mismatch Identification Table.

The Mismatch Identification Table is created by the SCE team members in Step 4 (➡page 5-3). The information to generate the form comes from the Proposed Project Profile and the Project Profiles submitted by the specific development organization. The team members compare the attributes of each project on the Proposed Project Profile to the attributes on the Project Profiles.

The Mismatch Identification Table is used by the SCE team in Step 4 to prepare the Experience Table. It is also used by the team members in Step 7 (➥page 5-31) as a guide to help select projects to investigate.

Mismatch Identification Table

Projects	Able	Baker	Charlie	Delta	Enigma	Fiesta	Result
Major Attributes							
Application Domain	0	0	1	0	0	0	
Product Type	1	1	1	0	0	0	
Size	0	0	0	0	0	0	Ps
Type of Work	1	1	0	1	1	0	
Subcontractors	1	1	1	1	1	1	
Minor Attributes							
Language(s)	0	1	1	0	0	0	
Target(s)	0	0	i	0	0	1	
Applicable Standards	0	1	1	0	0	0	
Customer	1	1	1	0	1	1	

0 = experience mismatch, 1 = experience match

Figure D-4: Sample Mismatch Identification Table

The Mismatch Identification Table lists the names of the attributes from the Proposed Project Profile form. Each row of the table corresponds to an attribute. Refer to Appendix C for a description of the attributes.

The form has a column for each project that is a candidate for evaluation. These columns show the result of comparing the attributes of each project that are listed on the Project Profile with the attributes of the product being developed,

as listed on the Proposed Project Profile. A "1" is placed in the table when the attributes match and a "0" when there is a mismatch.

The last column is the *Result* column. It shows the attributes of the product being procured where the development organization lacks experience. The abbreviation of the attribute is entered in the *Result* column if zeros are entered across the entire row. If there is at least one "1" in the row (i.e., there is previous experience) then the *Result* column is left blank.[1]

---

1.   On this form, the abbreviation "Ps" stands for "Product Size." This is used to represent the "Size" attribute.

Section D.5   Experience Table

The Experience Table is used to determine the attributes of the product to be developed for which any of the development organizations may lack previous experience. These attributes indicate areas of risk that should be given special attention during the evaluation. Figure D-5 is a sample Experience Table form.

Experience Table

Attribute Name	Offerors			
	Sigma Tech	Beverly Ind	Crystal City	Result
Major Attributes				
Application Domain				
Product Type		Pt		Pt
Size	Ps	Ps	Ps	Ps
Type of Work				
Subcontractors				
Minor Attributes				
Language(s)				
Target(s)				
Applicable Standards		Stds	Stds	Stds
Customer				

Figure D-5: Sample Experience Table

The Experience Table is created by the SCE team members in Step 4 (➡page 5-3). It is created by consolidating the *Result* columns of each of the Mismatch Identification Tables for each specific development organization an SCE will be applied to.[1]

The Experience Table is used by the SCE team members in Step 5 (➡page 5-13) to help select the subprocess areas that will be looked at during the evaluation. The subprocess areas selected for evaluation are referred to as critical subprocess areas; collectively these subprocess areas make up the Critical Subprocess Area List. The critical subprocess areas are the basis against which all development organizations are evaluated.

The Experience Table lists the names of the attributes from the Proposed Project Profile form. Each row of the table corresponds to an attribute. Refer to Appendix C for a description of the attributes.

The Experience Table also contains a column for each of the development organizations to be evaluated. Each column is a copy of the *Result* column from the Mismatch Identification Table for that development organization.

The last column is the *Result* column. It shows whether the development organizations, considered as a community, lack relevant experience in any of the attributes of the product being developed. Each row of the *Result* column contains the abbreviation for the attribute if the corresponding row of any other column contains an entry. Otherwise the entry is blank.

---

1.   On this form, the abbreviation "Ps" stands for "Product Size." This is used to represent the "Size" attribute.

Section D.6   Key Issue Table

The Key Issue Table is used to record the Critical Subprocess Area List that will be used to evaluate all development organizations. The table also indicates which of the critical subprocess areas should be given special attention for a specific development organization because of a lack of experience in that area (a Key Issue for that development organization). Figure D-6 shows a sample Key Issue Table.

The Key Issue Table is created by the SCE team in Step 5 (➥page 5-13). The information for the Key Issue Table comes from the tables provided as guidance in Appendix F, from the Target Product Profile created in Step 1 (➥page 4-15), from the Experience Table created in Step 4 (➥page 5-3), and from the Critical Subprocess Area List created in Step 5.

In Step 5, the team members select critical subprocess areas based on the experience of the development organizations and on whether the end user has experience with similar systems (*operational precedence*). The team also selects subprocess areas that represent basic processes that a development organization would need for any software development effort. This is referred to as a *nucleus capability*. Additional factors (such as the size of the undertaking) are used to extend and refine the list of subprocess areas. Collectively, these subprocess areas form the Critical Subprocess Area List. The Critical Subprocess Area List does not have a separate form—the Key Issue Table is used to document the list.

There is one Key Issue Table created for an SCE. The table will probably have multiple pages. The Key Issue Table is used in to develop the Key Issue Worksheet.

The Key Issue Table lists all the Key Process Areas (KPAs) included in the Target Process Capability. The critical

Key Issue Table

Critical Subprocess Area List	Offerors		
	Sigma Tech	Beverly Ind	Crystal City
Requirements Management			
Establish and maintain requirements baseline	Ps	Ps	Ps
Manage requirements-driven changes	Ps, *	Pt, Ps, *	Ps, *
Software Project Planning			
Develop estimates	Ps	Pt, Ps	Ps
Plan software activities	Ps	Ps	Ps
Make commitments	Ps, *	Ps, *	Ps, *
Software Project Tracking and Oversight			
Manage commitment changes			
Track progress	Ps, *	Pt, Ps, *	Ps, *
Take corrective action	Ps, *	Ps, *	Ps, *
Software Quality Assurance			
Plan SQA	Ps	Ps	Ps
Perform SQA	Ps, *	Ps, *	Ps, *
Address noncompliance	Ps, *	Pt, Ps, *	Ps, *
Software Configuration Management			
Create software work products baseline	*	Pt, *	*
Control changes	Ps, *	Pt, Ps, *	Ps, *

Figure D-6: Sample Page of a Key Issue Table

subprocess areas are listed under the KPA with which they are associated. There will be at least one subprocess area selected for each KPA in the Target Process Capability.

The table also contains a column for each development organization. These columns show why a subprocess area was selected and whether the subprocess area needs to be given special attention for a specific development organization. (This indicates that the subprocess area is a key issue for the organization.) The following criteria are used to indicate the relationships between the development organizations and the subprocess areas:

- If a subprocess area was selected because of a lack of experience for a particular project attribute, as indicated in the Experience Table, the abbreviation for the attribute is entered in the column for each development organization that lacked experience.[1]

- If the subprocess area was selected because the end user lacks operational precedence with similar systems (as indicated on the Target Product Profile), the column contains the abbreviation "Op".

- If the subprocess area was selected because it is associated with a *nucleus capability*, the column contains an asterisk ("*").

- If there is no entry in the column, it means the subprocess area was selected because of a lack of experience elsewhere in the development organization community, or added to the list because of team judgment. This subprocess area will be investigated, but the team may decide to spend more time on other subprocess areas.

---

1. On this form, the abbreviation "Ps" stands for "Product Size." This is used to represent the "Size" attribute.

Section D.7   Validation Worksheet

The Validation Worksheet contains the topics that will be explored during the site visit for a specific development organization. The worksheet is used to record the team's consensus on the data they have collected for each topic. Figure D-7 below shows a sample Validation Worksheet.

The Validation Worksheet is prepared by the SCE team. In Step 6 (➡page 5-26), the team members create a set of Validation Worksheets. One worksheet is created for each subprocess area in the Critical Subprocess Area List, as documented on the Key Issue Table. A copy of the set of worksheets is made to be used for each development organization. In Step 10 (➡page 5-41), the team members add topics for each subprocess area to the worksheets. The consolidated topic list is created in Step 9 (➡page 5-39).

In Step 11 (➡page 5-44), the Validation Worksheets are used to generate interview questions. The Validation Worksheets are used throughout the site visit to record when consensus has been reached on a topic and to determine what topics need to be pursued in follow-on interviews and document reviews.

The top of each page of the form contains the name of the KPA and subprocess area, and a space for the name of each project being evaluated. The names of the projects are preceded by a letter that is used to identify the information for a project.

The form contains a row for each topic associated with the subprocess area. The topics are listed in the first column.

The next four columns are subdivided into rows for each of the projects being evaluated. The first of these columns contains the letter to indicate which project the information in the row is associated with. The other three columns are used to record whether the team reaches a

| Carnegie Mellon University
Software Engineering Institute | | | **SCE Validation Worksheet** | | |

Projects: A. <u>Able</u>   B. <u>Baker</u>   C. <u>Charlie</u>   D. <u>_____</u>

Software Project Planning Develop estimates	Project	Explore Interview	Doc Review	Consolid Interview	Organization
Comments: look at both cost and size estimates	A				
	B				
	C				
	D				
organizational policies	A				
	B				
	C				
	D				
organizational structures	A				
	B				
	C				
	D				
training	A				
	B				
	C				
	D				

List of people interviewed:

Figure D-7: Sample Page of a Validation Worksheet

consensus on a topic for a project as a result of exploratory interviews, documentation reviews, or consolidation interviews.

The last column of each row is used to record the composite finding on the topic for the organization.

Section D.8   SCE Questionnaire Worksheet

Each development organization completes a questionnaire for the projects that the team may evaluate. Usually questionnaires are completed for all of the six to eight projects that are candidates for evaluation. (These are same projects listed on the Project Profile form.) In some cases, the questionnaire will only be required for the 3 to 4 projects selected for evaluation.

The questionnaire is used to collect information about the software development processes used on the projects that will be evaluated. The questionnaire provides an initial data input to the SCE team about the processes in use.

The SCE Questionnaire Worksheet is used to summarize the questionnaire responses submitted by a development organization. Figure D-10 is a sample page from an SCE Questionnaire Worksheet. The example questions shown on the form are drawn from the CMM based questionnaire.

The SCE Questionnaire Worksheets are prepared in Step 8 (➡page 5-34). A worksheet is prepared for each development organization. The worksheet will have multiple pages. The SCE team members copy the information from the questionnaire for the projects selected for evaluation. The worksheets make it possible to compare results from all projects and to map question responses to subprocess areas to be investigated.

The SCE Questionnaire Worksheets are used by the SCE teams in Step 8 in the preparation of the Key Issue Worksheet. The SCE Questionnaire Worksheets are reviewed for inconsistencies and anomalies which indicate critical subprocess areas that should receive special attention for a specific development organization.

Figure D-8: Sample Page of a Questionnaire Worksheet

The top line of the SCE Questionnaire Worksheet contains the names of the projects that are being evaluated. The names of the projects are preceded by a letter that is used to indicate which project a response corresponds to.

The rows of the SCE Questionnaire Worksheet are grouped by KPA and subprocess area. The name of the KPA is listed in a box at the top of the page. An abbreviation for the subprocess area is listed at the top of the group of rows. The subprocess area name is abbreviated as the KPA abbreviation and the number of the corresponding KPA goal. There may be more than one subprocess area group on a page.

Under each subprocess area heading are questions that are related to the subprocess area. The questions are listed in boxes in the left column of the form. The question number is shown in the upper left corner of the box. The maturity level that the question corresponds to is listed in the upper right corner of the box.

The next two columns are subdivided into rows for each of the projects evaluated. The first of these columns contains a letter to indicate which project the response is for. The second of the two columns is used to record the responses to the questions from the questionnaire.

The last column of each row is used to record any comments from the questionnaire.

Section D.9   Key Issue Worksheet

The Key Issue Worksheet is used to collect all of the information available about a development organization in one place so the team can determine the relative amount of time to spend investigating each of the critical subprocess areas during the site visit.

The Key Issue Worksheet also supports analysis of the Questionnaire Worksheets prepared for each development organization. This analysis may indicate subprocess areas that should receive special attention during the evaluation because of apparent inconsistencies or anomalies in the questionnaire responses. An anomaly occurs when the response to one question by one or more projects is different. An inconsistency occurs when responses to two questions for the same project are apparently in conflict.

Taken by itself, any questionnaire is limited by the focus of the questions asked. However, the standard SEI questionnaires can point the SCE team to a specific part of the critical subprocess area by identifying anomalies and inconsistencies. Figure D-10 shows a sample Key Issue Worksheet.

The Key Issue Worksheet is created by the SCE team in Step 8 (➡page 5-34). The information comes from the Key Issue Table and from the Questionnaire Worksheet.

The Key Issue Worksheet is used in Step 9 (➡page 5-39) to develop the topic lists that will guide the interviews and document reviews for a specific development organization.

The *Critical Subprocess Areas* column of the Key Issue Worksheet is taken from the Key Issue Table (Figure D-6). It lists the KPAs and the critical subprocess areas that will be investigated.

The second column shows why each subprocess area is important with regard to the specific development organization. It is the same as the column from the Key

Key Issue Worksheet

Critical Subprocess Areas	Sigma Tech	Projects Able	Projects Baker	Charlie
Requirements Management				
Establish and maintain requirements baseline	Ps			
Manage requirements-driven changes	Ps, *			
Software Project Planning				
Develop estimates	Ps			Inc: est. training
Plan software activities	Ps			
Make commitments	Ps, *			
Software Project Tracking and Oversight				
Manage commitment changes		Customer I/F	Customer I/F	
Track progress	Ps, *			
Take corrective action	Ps, *	issue trking		issue trking
Software Quality Assurance				
Plan SQA	Ps			
Perform SQA	Ps, *	An:CDRLs	An:CDRLs	An:CDRLs
Address noncompliance	Ps, *			
Software Configuration Management				
Create software work products baseline	*			
Control changes	Ps, *	CCB	CCB	CCB

Figure D-9: Sample Page of a Key Issue Worksheet

Issue Table that shows the experience mismatches for the development organization (➡page D-14).

There is also one column for each of the projects selected for evaluation. This column is used to record the results of reviewing the SCE Questionnaire Worksheet for inconsistencies and anomalies.

Anomalies and inconsistencies are recorded in the rows corresponding to the subprocess areas to which the question applies. When an anomaly or inconsistency is found, an abbreviated summary of the response is recorded. It is sometime handy to annotate the question numbers as well. An example of an anomaly and an inconsistency follow.

Anomaly

Consider the question within the Software Quality Assurance key process area:

- Do SQA activities provide objective verification that *software products* and activities adhere to applicable standards, procedures, and requirements?

If this question is answered "yes" for three projects and "no" for one of the selected projects then that can be considered to be an *anomaly* in the organization in that SQA does not appear to be the same for all projects. This is what the "An: CDRL" entry refers to in the "perform SQA" subprocess area in Figure D-10 (➡page D-28).

Inconsistency

Consider the following questions within the Software Configuration Management key process area:

- Does the project follow a documented procedure to control changes to configuration items/units?

- Are project personnel trained to perform the software configuration management activities for which they are responsible?

If one project responds "no" to the first question, and "yes" to the second question, then the team may consider this an inconsistency in that they may wonder about the quality and content of the training if there is no documented procedure to guide the change control activities.

Section D.10   Interview Worksheet

The Interview Worksheet is used as a guide for an interview with a specific person. It contains the KPAs and subprocess areas that are to be investigated for that person with questions that will be asked. The worksheet is used to record the responses to the interview questions. Figure D-10 shows a sample Interview Worksheet.

The Interview Worksheets for the exploratory interviews are prepared by the SCE team in Step 11 (➡page 5-44). Additional Interview Worksheets may be prepared during the team caucus sessions at the site interview as the need for follow-on interviews is determined. The information for the Interview Worksheets comes from the Validation Worksheets.

The Interview Worksheets are used by the team members to record the interview responses throughout the site visit.

The Interview Worksheet contains two columns. The first column contains the questions to be asked and any notes, such as types of documentation to request, that may be needed during the interview. This column also contains the KPA and subprocess area that the question is associated with. The second column is used to record the responses to the questions.

The header for the Interview Worksheet contains the position of the person being interviewed, the name of the person, and the time of the interview.

Interview Worksheet	
Interviewee's Name:	Date:
Position:	Time:

Question	Response

Requirements Management
   Establish and maintain requirements
   baseline
      What is your role in maintaining the
      baseline requirements?
 	
      How is the requirements baseline
      managed?
 	
Possible documents:
   policy and procedures for a CCB
   position description	

Requirements Management
   Manage requirements driven changes
      How are changes resulting from new
      requirements managed?
 	
      How are changes tracked?
 	
Possible documents:
   CCB minutes,
   revised size and cost estimates,
   traceability matrix	

<KPA>
  <subprocess area>
      question 5
      question 6
<possible document types>	

Figure D-10: Sample Page of an Interview Worksheet

Appendix E   Blank Forms

This appendix provides blank copies of the forms used throughout this document. The blank forms may be used as reproduction masters.

The forms included here are based on the ones used during the SCE team training; in some cases they have been resized to fit in this document better.[1] These forms are conceptual in nature; they indicate information needed to conduct an SCE, but they are not mandatory.

Blank copies of the following forms are shown in this appendix:

---

1. The terminology of the forms is acquisition oriented because that was the focus of the initial training, and is still the primary use of the SCE method. For example, "offeror" is used for "development organization" on some of the forms.

Section E.1   Target Product Profile

The following is a blank form for creating the Target Product Profile. The Target Product Profile is created in Step 1 (➥page 4-15).

Target Product Profile

Attributes	RFP Development

Major Attributes
 	
Application Domain
 	
Product Type
 	
Size
   Contract Duration
   Software Team Size
   Estimated Software Size (KSLOC)
 	
Type of Work
 	
Operational Precedence
 	
Minor Attributes
 	
Language(s)
 	
Target
 	
Applicable Standards
 	
Customer	

Section E.2   Proposed Project Profile

The following is a blank form for creating the Proposed
Project Profile. The Proposed Project Profile is created by
the development organization prior to Step 4 (➡page 5-3).

Proposed Project Profile

Attributes	Proposed Development

<u>Major Attributes</u>
 	
Application Domain
 	
Product Type
 	
Size
    Contract Duration
    Software Team Size
    Estimated Software Size (KSLOC)
Type of Work
 	
Subcontractors
 	
<u>Minor Attributes</u>
 	
Language(s)
 	
Target
 	
Applicable Standards
 	
Customer
 	
Host Development System
 	
Configuration Management Tool	

Section E.3   Project Profiles

The following is a blank form for creating the Project Profiles. The Project Profiles are created by the development organization prior to Step 4 (➡page 5-3).

Project Profiles

Project			
Major Attributes			
Application Domain			
Product Type			
Size			
Contract Duration			
Software Team Size			
Estimated Software Size (KSLOC)			
Type of Work			
Subcontractors			
Minor Attributes			
Language(s)			
Target			
Applicable Standards			
Customer			
Host Development System			
Configuration Management Tool			
Schedule Data			
Current Phase			
Current Month			
Start			
Design Ends			
Coding Ends			

Section E.4   Mismatch Identification Table

The following is a blank form for creating the Mismatch Identification Table. The Mismatch Identification Table is created in Step 4 (➥page 5-3).

Mismatch Identification Table

Projects							Result
Major Attributes							
Application Domain							
Product Type							
Size							
Type of Work							
Subcontractors							
Minor Attributes							
Language(s)							
Target(s)							
Applicable Standards							
Customer							

0 = experience mismatch, 1 = experience match

Section E.5   Experience Table

The following is a blank form for creating the Experience Table. The Experience Table is created in Step 4 ().

Experience Table

Attribute Name	Offerors				Result
Major Attributes					
Application Domain					
Product Type					
Size					
Type of Work					
Subcontractors					
Minor Attributes					
Language(s)					
Target(s)					
Applicable Standards					
Customer					

Section E.6   Key Issue Table

The following is a blank form for creating the Key Issue Table. The Key Issue Table is created in Step 5 (➡page 5-13).

Key Issue Table

Critical Subprocess Area List	Offerors			

Section E.7   Validation Worksheet

The following is a blank form for creating the Validation
Worksheet. The Validation Worksheet is created in Step 6
(➡page 5-26).

Carnegie Mellon University
Software Engineering Institute

SCE Validation Worksheet

Projects: A. _____   B. _____   C. _____   D. _____

	Project	Explore Interview	Doc Review	Consolid Interview	Organization
	A				
	B				
	C				
	D				
	A				
	B				
	C				
	D				
	A				
	B				
	C				
	D				
	A				
	B				
	C				
	D				

List of people interviewed:

Section E.8   Key Issue Worksheet

The following is a blank form for creating the Key Issue Worksheet. The Key Issue Worksheet is created in Step 8 (➥page 5-34).

Key Issue Worksheet

Critical Subprocess Areas		Selected Projects				

Section E.9   Interview Worksheet

The following is a blank form for creating the Interview Worksheets. The Interview Worksheet is created in Step 11 (➡page 5-44).

Interview Worksheet	
Interviewee's Name:	Date:
Position:	Time:
Question	**Response**

Section E.10   Document Review Checklist A

The following is a blank form for Checklist A which is used during the initial document review, Step 14 (➡page 6-15).

Checklist A
(for each policy and procedure)

Name: ☐ Policy

                                               ☐ Procedure

Information sought	Found	Comment
1. Source quality Who originated? Is a review record visible? Is the approval record visible? Is the audience identified? Is the date it came in force visible?		
2. Version data Does it have a current version number? Is the version number it supersedes visible?		
3. Content Is a scope defined? Is there a purpose statement?		
4. Glossary Are there definitions of terms and/or acronyms?		
5. Relationships If policy, is there a list of procedures identified? If procedure, is there a defined tailoring mechanism for projects to use?		
6. Configuration management control Is there a configuration management responsibility for these policies and procedures?		

Figure E-1: Document Review Checklist A

Section E.11   Document Review Checklist B

The following is a blank form for Checklist B which is used
during the initial document review, Step 14 (➠page 6-15).

Checklist B
Organization's Policies and Procedures
Related to Critical Subprocess Areas

Critical Subprocess Areas	Applicable Organization Documents Plus Comments

Figure E-2: Document Review Checklist B

Section E.12   Document Review Checklist C

The following is a blank form for Checklist C which is used during the initial document review, Step 14 (➡page 6-15).

Checklist C
Summary of Organizational Documents by Key Process Area

Key Process Area	Comment
Requirements Management	
Software Project Planning	
Software Project Tracking and Oversight	
Software Subcontract Management	
Software Configuration Management	
Software Quality Assurance	
Organization Process Focus	
Organization Process Definition	
Software Product Engineering	
Integrated Software Management	
Intergroup Coordination	
Peer reviews	
Training Program	

Figure E-3: Document Review Checklist C

Appendix F   Subprocess Area Selection Tables

This appendix contains information used to help SCE teams select critical subprocess areas for evaluation. Critical subprocess areas are selected in Step 5, "Create Critical Subprocess Area List" (➡page 5-13).

There are several things the team should consider when selecting subprocess areas. General factors that should be considered in selecting critical subprocess areas include the following:

- What processes would an organization need to manage the aspects of the project which are new to the organization?

- If the product being developed is new to the end user, what processes will the development organization need to manage the anticipated requirements changes?

- What are the basic processes that a development organization would need for any software development effort?

This appendix contains tables the teams can use to help select critical subprocess areas. The tables were created by SCE project members at the SEI for guidance only. SCE teams are expected to use their experience and judgement to select critical subprocess areas based on the requirements of the particular development.

There are two sets of tables, respectively based on:

- The size of the development undertaking (➡page F-3).

- Mismatches indicating a lack of experience either in the development organization or the end user of the system (➡page F-5).

The size of the development undertaking can be used to select subprocess areas as critical, as described in Section F1.

Section F2 (➡page F-5), contains information that can be used two ways. First, the project profiles and the proposed project profile may indicate that a particular subprocess area is significant for the product to be acquired because of lack of experience in some attribute associated with developing the product to be acquired. These tables also indicate a recommended nucleus capability of subprocess areas that should be considered for every SCE.

Section F1   Selecting Critical Subprocess Areas Based on Size of the Development Undertaking

This section contains tables that show the relationship between the number of levels of management within the development undertaking and candidate critical subprocess areas. The size of the development undertaking is indicated by the proposed project profile; information about the levels of management required for the project is found by examining information provided about the organizational structure. Table F-1 shows the relationship between subprocess areas and the size of the development undertaking.

Size of Development Undertaking	KPA	Subprocess area
Major Undertaking (Software Manager has reports from two or more second line software managers)	Software Project Planning	Develop documented estimates.
		Obtain agreement on planned commitments.
	Software Configuration Management	Identify selected software work products for a baseline, which is controlled and made available.
		Control changes to software baselines.
	Intergroup Coordination	Obtain agreement by affected groups on commitments between engineering groups.
	Integrated Software Management	Define project's software process by tailoring the organization's standard software process.
	Peer Reviews	Identify and remove defects in software work products.
Medium Undertaking (Software manager has reports from two or more supervisors)	Software Quality Assurance	Verify adherence of activities and products to applicable standards
		Address non-compliance issues.
	Software Project Planning	Plan, and document software activities and commitments.
	Peer Reviews	Plan peer review activities.
	Software Project Tracking and Oversight	Take and manage corrective actions to reduce variance from plans.
		Obtain agreement on commitment changes.
Small Undertaking (Software manager has reports only from software leads)	Software Project Tracking and Oversight	Track progress against software plans.
	Software Quality Assurance	Plan software quality assurance activities.
		Communicate SQA results.

**Table F-1: Critical Subprocess Areas Based on
Size of the Development**

Section F2   Selecting Critical Subprocess Areas Based on Experience Mismatches

The entries in this table represent consensus judgment from a group of experienced practitioners at the SEI. Selection of subprocess areas using these tables should be tempered by team judgment, experience, and detailed knowledge of the planned development.

How To Read the Tables

This section contains a table for each key process area (KPA) in the *Repeatable* and *Defined* levels. The tables contain the following columns.

KPA and Subprocess Areas Column

Each row under this column corresponds to a KPA or a subprocess area associated with the KPA. The KPAs are indicated by boldface type.

Major Attributes Columns (ApD, Pt, Ps, Tw, and Sub)

A black square (■) in the column for an attribute indicates that the subprocess area listed in that row may be important to the development organization for managing the risk associated with a lack of experience relative to that attribute. These columns correspond to the five major attributes from the Experience Table created in Step 4. The Experience Table shows where any of the development organizations may lack experience with regard to some attribute of the new project. Refer to Appendix C for a definition of each attribute.

Operational Precedence (Op) Column

A black square (■) in this column indicates that the subprocess area listed in that row may be important for managing the level of requirements changes which may be anticipated if end users do not have experience with

similar products. The **Op** column corresponds to the *operational precedence* attribute from the Target Product Profile developed by the sponsor. This attribute indicates the degree to which the product being developed may be new to the end user. Refer to Appendix C for a definition of the *operational precedence* attribute.

Nucleus Capability (*)
Column

A black square (■) in this column indicates that the subprocess area listed in that row is part of the recommended *nucleus capability*. Nucleus capability refers to a basic set of processes which are needed for almost any software development.

Repeatable Level Key Process Areas (KPAs)

Key to Abbreviations

ApD	Application Domain	**Tw**	Type of Work	**Op**	Operational Precedence
Pt	Product Type	**Sub**	Subcontracting	*****	Nucleus Capability
Ps	Size				

Repeatable Level Key Process Areas

Key Process Areas and Subprocess Areas	Major Attributes					Op	*
	ApD	Pt	Ps	Tw	Sub		
Requirements Management							
Establish and maintain requirements baseline	■		■	■	■	■	
Manage requirements-driven changes	■	■	■	■	■	■	■
Software Project Planning							
Develop estimates	■	■	■	■			
Plan software activities			■	■			
Make commitments			■	■	■		■
Software Project Tracking and Oversight							
Track progress			■	■			■
Take corrective action	■	■	■	■	■		■
Manage commitment changes							

Table F-2: Selection Table for Repeatable Level Subprocess Areas

Key to Abbreviations

ApD	Application Domain	**Tw**	Type of Work	**Op**	Operational Precedence
Pt	Product Type	**Sub**	Subcontracting	*****	Nucleus Capability
Ps	Size				

Repeatable Level Key Process Areas (Continued)

Key Process Areas and Subprocess Areas	Major Attributes					Op	*
	ApD	Pt	Ps	Tw	Sub		
Software Subcontract Management							
Select subcontractors					■		
Establish and maintain commitments		■		■	■		■
Maintain communications							
Track progress				■	■		■
Software Quality Assurance							
Plan SQA			■	■			
Perform SQA			■	■			■
Communicate results					■		
Address noncompliance	■	■	■		■		■
Software Configuration Management							
Plan SCM				■			
Create software work products baselines	■	■		■	■	■	■
Control changes	■	■	■		■	■	■
Report status							

Table F-2: Selection Table for Repeatable Level Subprocess Areas (Continued)

Defined Level Key Process Areas (KPAs)

Key to Abbreviations					
ApD	Application Domain	**Tw**	Type of Work	**Op**	Operational Precedence
Pt	Product Type	**Sub**	Subcontracting	*	Nucleus Capability
Ps	Size				

Defined Level Key Process Areas

Key Process Areas and Subprocess Areas	Major Attributes					Op	*
	ApD	**Pt**	**Ps**	**Tw**	**Sub**		
Organization Process Focus							
Coordinate software process activities	■		■	■	■		
Assess software processes used							■
Plan SPI	■			■			
Organization Process Definition							
Provide standard process							
Retain software process information							■
Software Product Engineering							
Build software	■	■	■	■	■	■	■
Ensure consistency							

Table F-3: Selection Table for Defined Level Subprocess Areas

Key to Abbreviations

ApD	Application Domain	**Tw**	Type of Work	**Op**	Operational Precedence
Pt	Product Type	**Sub**	Subcontracting	*****	Nucleus Capability
Ps	Size				

Defined Level Key Process Areas (Continued)

Key Process Areas and Subprocess Areas	Major Attributes					Op	*
	ApD	Pt	Ps	Tw	Sub		
Integrated Software Management							
Define project process	■	■	■	■			
Manage according to process							■
Intergroup Coordination							
Agree on customer's requirements	■	■	■	■	■	■	
Coordinate intergroup commitments			■	■			
Manage intergroup issues							■
Peer Reviews							
Plan peer review							■
Identify and remove defects	■	■	■	■			■
Training Program							
Plan training							■
Provide training							
Receive necessary training	■	■	■	■			■

Table F-3: Selection Table for Defined Level Subprocess Areas (Continued)

Managed Level Key Process Areas (KPAs)

Key to Abbreviations

ApD	Application Domain	**Tw**	Type of Work	**Op**	Operational Precedence
Pt	Product Type	**Sub**	Subcontracting	*****	Nucleus Capability
Ps	Size				

Managed Level Key Process Areas							
Key Process Areas and Subprocess Areas	**Major Attributes**						
	ApD	**Pt**	**Ps**	**Tw**	**Sub**	**Op**	*****
Quantitative Process Management							
Plan QPM							
Control process quantitatively	■	■	■				
Establish organization's process capability							
Software Quality Management							
Plan quality management							
Define software quality goals							
Track quality progress	■	■	■				

Table F-4: Selection Table for Managed Level Subprocess Areas

Optimizing Level Key Process Areas (KPAs)

Key to Abbreviations

ApD	Application Domain	**Tw**	Type of Work	**Op**	Operational Precedence
Pt	Product Type	**Sub**	Subcontracting	*****	Nucleus Capability
Ps	Size				

Optimizing Level Key Process Areas

Key Process Areas and Subprocess Areas	Major Attributes					Op	*
	ApD	Pt	Ps	Tw	Sub		
Defect Prevention							
Plan defect prevention							■
Identify defect causes							
Eliminate defect causes	■	■	■	■			
Technology Change Management							
Plan technology changes	■	■					
Evaluate new technologies	■	■					
Adopt new technology							
Process Change Management							
Plan process improvement							
Empower everyone							
Continuously improve							

Table F-5: Selection Table for Optimized Level Subprocess Areas

Appendix G   Look-For Tables

The tables in this appendix are provided to help SCE teams probe the critical subprocess areas selected for evaluation.

This appendix has three sections:

Section	Page number
G.1 - Agent, Artifact, Relationship Examples	page G-3
G.2 - Standard Look-for Table	page G-29
G.3 - Probing Guides for Key Process Areas	page G-36

Look-for tables provide guidance SCE teams can use to probe and collect objective evidence about a development organization's implementation of a subprocess area. Based on the evidence gathered, the SCE team judges whether the development organization has achieved the goal the subprocess area corresponds to.

Look-for tables serve two general purposes–they give examples of topics to probe for and they can help SCE teams make judgments about the information they collect. The Look-for tables give examples of topics to probe for by providing clues about who to interview, what might be asked, what objective evidence to look for, and what documents to review. They can help SCE teams consolidate the information they collect by associating specific activities that might be observed with the appropriate subprocess areas and KPAs; this is critical for making accurate judgments.

The Look-for tables are divided into three parts:

- Examples of Agents, Artifacts, and Relationships,
- Standard Look-for Table,

- Probing Guides for Key Process Areas.

The Examples of Agents, Artifacts, and Relationships are organized by KPA. There is one table for each subprocess area in the KPA. These tables provide information to help teams discover the specific activities that a development organization has chosen to implement their software processes. SCE teams can use this information to help refine the topics selected for investigation, and to help plan for interviews and document reviews.

There is one Standard Look-for Table which is applicable to all subprocess areas. It provides general guidance to help the teams determine whether the goal of the KPA has been satisfied with respect to the common features of the CMM. These guidelines may be used both to help focus the investigation and to help the team make judgements about the information collected.

The Probing Guides are organized by KPA and subprocess area. There is one table for each subprocess area in the KPA. The probing guides help the teams to determine if the KPA goals have been satisfied. They provide information about the types of activities associated with each subprocess area. These guidelines can be used to help focus the investigation for specific subprocess areas, and also to help the team make judgements about the data collected.

The Standard Look-for Table and the Probing Guides provide an extensive cross-reference between the subprocess areas used by SCE teams and specific activities and features described in CMM v1.1 [Paulk 93a] and the companion volume, *Key Practices of the Capability Maturity Model, Version 1.1* [Paulk 93b]. The Standard Look-for Table is equally applicable to all subprocess areas; the Probing Guides are unique to each subprocess area.

Section G.1   Agent, Artifact, Relationship Examples

How these tables should be used

The agents, artifacts, and relationships described should be viewed as conceptual in nature. All defined process must have defined agents, artifacts, and relationships. During SCE data collection, the intent is to identify the the agents and artifacts in the terms the organization uses, within the context that they are used. This conceptual notion of agents, artifacts, and relationships provides a way to understand the organization's processes which is free from preconceived notions about organizational structure or responsibilities assigned to particular roles.

The Examples of Agents, Artifacts, and Relationships are provided to help SCE refine topics, and to help plan their interviews and document reviews. The example agents indicate the types of people that might be interviewed about a particular topic. The example artifacts indicate the types of documents that might be reviewed. The example relationships provide clues to the types of activities teams might look for.

These tables should not be used as a checklist.

As a case in point, consider the first table for the Software Project Planning KPA on page 6. The subprocess area is "Develop estimates". The first set of agents listed includes senior software engineers, software managers, system engineers, and testing managers. The artifacts associated with them are "allocated requirements" and an "estimate package". The examples of relationships includes "Senior software engineers have estimated the software components to be built and included the type of effort for each component in the Estimate Package." A particular development organization may have a cost estimation shop manned by "cost estimators" (agents) that prepare a "work breakdown structure" (an artifact) based on the

"specification". Using the example in the table as a checklist might cause a team to miss well-defined processes in use at this organization.

Table Format

The tables are self explanatory, except to mention that the agents and artifacts are grouped into related sets. In the example cited above, the agents are senior software engineers, software managers, system engineers, and testing managers. Any or all of the agents in this group are likely to be associated with the artifacts called allocated requirements and estimate package.

Definitions

Agents are roles within the organization's operation that carry specific responsibility (e.g., test director, lead engineer, project planner, senior manager). An agent may perform an activity, provide an input for the activity, receive the output, define how the activity is to be performed, or verify that the activity is performed.

Artifacts (or objects) are the work products which are part of the activity. Artifacts may be either process artifacts or product artifacts. Context determines whether a particular artifact is a process artifact or a product artifact (e.g., a CM Plan will be a product artifact as a deliverable, and a process artifact from the context of the software product development).

Relationships indicate the roles agents and artifacts play in the activities and serve as examples of possible activities that teams might look for.

Key Process Area: Requirements Management

Subprocess area: Establish and maintain requirements baseline
Action: Establish and maintain a baseline of agreed-upon requirements allocated to software

Examples of Agents	*Examples of Artifacts*
Software Engineers	Allocated baseline (e.g., traceability matrix, human
Software testers	interface)
System engineers	Project schedule (e.g., milestones, inchstones)
Customer	
System engineers	Allocated baseline
Hardware engineers	Prototyping results
Software engineers	Benchmark results

Examples of relationships between agents and artifacts

- The stakeholders[a] accept the allocated baseline and project schedule.
- The stakeholders ensure the allocated baseline contains a traceability matrix (and a section on human interface if applicable).
- System, hardware, and software engineers use prototyping and benchmarking techniques to clarify the allocated baseline.

a. The stakeholders might be any or all of the agents listed.

Subprocess area: Manage requirements-driven changes
Action: Manage requirements-driven changes

Examples of Agents	*Examples of Artifacts*
Software managers	Project schedule, a᠁᠁ plans
Testing managers	Work authorizatiᴜᴜ᠁
	Changes to the allᴜᴜ ᴜᴜ baseline
Software engineers	Software Development Files
Software testers	Software development plan
Software supervisors	Work products
System engineers	Changes to the allocated baseline
	Software Trouble Report(s)

Examples of relationships between agents and artifacts

- Software and testing managers know the work authorizations and project plans are consistent with the current allocated baseline.
- First line supervisors, software developers, and software testers maintain a file which journals inputs, events, intermediate products, and outputs that show the work products are consistent with the changes to the allocated baseline.

Key Process Area: Software Project Planning

Subprocess area: Develop estimates
Action: Develop documented estimates

Examples of Agents[a]	Examples of Artifacts
Senior software engineers	Allocated requirements
Software managers	Estimate package (e.g., bases of estimate with
System engineers	assumptions, task descriptions, labor spread over
Testing managers	schedule)
Senior software engineers	Productivity coefficients and parameters
Software managers	Historical database
Accounting staff	Estimation tool
	Cost package

Examples of relationships between agents and artifacts

- Senior software engineers have estimated the software components to be built and included the type of effort for each component in the Estimate Package.
- Senior software engineer analysis of the development work, and use of level of effort historical data, are recorded on Task Descriptions, a Basis of Estimate for each task, together with an assessment of type of labor required over the Proposed Schedule.
- Senior software engineers are involved in selecting the parameters that are appropriate for the development of the cost package.
- Accounting staff has an estimation tool and/or an equivalent process to translate size estimates into cost estimates.

a. Agents and artifacts are grouped in sets that indicate possible relationships.

Key Process Area: Software Project Planning (cont'd)

Subprocess area: Plan software activities
Action: Plan and document software activities and commitments

Examples of Agents	*Examples of Artifacts*

Project planners
Software manager
Software supervisors
Work Force
	Planning input (e.g., contract, proposal, statement of
	   work, allocated baseline, CDRL schedule, size and
	   cost packages, project schedule set)
	Project planning tools
	Software Development Plan

Project management
Software manager
	Software Development Plan (e.g., development
	   resources and facilities, software components,
	   planned target hardware utilization)
	Risk plan
	Engineering Change Proposals (Change Requests)
	Non-deliverable project work
	Project schedule set
	Work authorization sheets

Examples of relationships between agents and artifacts

- Project planners use all the information from the size and cost estimation packages and the awarded contract to create a project schedule for each working group's activities that requires separate management.
- Project management checks that work authorizations are based upon the work tasks identified in the project schedule set.
- Project and software management keep the Software Development Plan, work authorization sheets and project schedule set updated to the status of the development work.

Subprocess area: Make commitments
Action name: Obtain agreement on planned commitments

Examples of Agents	*Examples of Artifacts*
Software manager Software supervisors Work force	Work authorization description of work Work schedule

Examples of relationships between agents and artifacts

- Software manager and software supervisors use formal acceptance and commitment for all schedules to pace the work of a work force.

Key Process Area: Software Project Tracking and Oversight

Subprocess area: Track progress
Action: Track progress against software plans

Examples of Agents	*Examples of Artifacts*
Software managers	Schedules (e.g., Gantt charts and inchstones)
Software supervisors	Work breakdown structure accounts
Work force	Software build plan
Senior management	Weekly reports (includes change activity)
Software managers	Monthly project report (includes change activity)
Personnel with progress responsibility	Project milestones
	Variance report(s)

Examples of relationships between agents and artifacts

* The work force collects level of effort expended into work breakdown structure accounts and tracks the software build plan and the schedule inchstones.
* Managers, supervisors, and work force personnel all use schedules against which to measure progress within their responsibility.
* Managers, supervisors, and work force personnel all make regular progress status reports, which are reviewed by their managers.

Subprocess area: Take corrective action
Action: Take and manage corrective actions to reduce variance from plans

Examples of Agents	*Examples of Artifacts*
Software supervisors	Correction activities within Trouble Reports
Work force	Action items
Software manager	Library file for all types of corrective actions
Software manager	Corrective action profiles
	(Risk plan if necessary)
Senior management	Monthly project reports

Examples of relationships between agents and artifacts

* Supervisors and work force track progress against action items.
* The work force enters all action items that arise into library records.
* Managers analyze the current profile of action item activities regularly for corrective action purposes.
* Senior management reviews monthly project reports, and provides assistance where necessary.

Key Process Area: Software Project Tracking and Oversight (cont'd)

Subprocess area: Manage commitment changes
Action: Obtain agreement on commitment changes

Examples of Agents	*Examples of Artifacts*
Software engineers	Engineering Change Proposal (Change Request)
Software supervisors	Project plan adjustment
Software Testers	
Software manager	Work authorization sheet
	Schedules
	Technical Interchange Meeting minutes

Examples of relationships between agents and artifacts

- Software engineers and supervisors put major software product changes through full estimation, cost, and planning activities.
- Managers cause adjustments to the work authorization sheets and schedules, and go through a commitment process with both managers and work force affected for all major software product changes.

Key Process Area: Software Subcontract Management

Subprocess area: Select subcontractors
Action: Select qualified subcontractors

Examples of Agents	*Examples of Artifacts*
Subcontract manager	Subcontract including Statement of Work
Software manager	Guidelines for selection of software subcontractors

Examples of relationships between agents and artifacts

* The Software manager was part of the process to select the subcontractor.

Subprocess area: Establish and maintain commitments
Action: Obtain and maintain agreement by contractor and subcontractor for mutual commitments

Examples of Agents	*Examples of Artifacts*
Subcontract manager	Subcontract (including Statement of Work)
Software manager	Interface Requirements Specification
SQA engineer	SQA plan
	Software Development Plan
	Project management plan

Examples of relationships between agents and artifacts

* The software project's management personnel agree to the interface between the subcontracted part of the product and the rest of the product.
* The software manager verifies the subcontracted part is identified in an Interface Requirements Specification.
* SQA has an assigned role for the subcontract.
* The managers make sure that the subcontract's statement of work contains the allocated system requirements appropriate to the subcontracted work.

Subprocess area: Maintain communications
Action: Maintain communications with subcontractor

Examples of Agents	*Examples of Artifacts*
Software manager	Technical Interchange Meetings
Software supervisors	

Examples of relationships between agents and artifacts

* All stakeholders in the subcontract are involved in the Technical Interchange Meetings or the equivalent.

Subprocess area: Track progress
Action: Track subcontractor progress

Examples of Agents	*Examples of Artifacts*
Senior management	Subcontract progress reports
Subcontract manager	Subcontractor SQA report
Software manager	Subcontractor SCM report
	SCM reports
	SQA reports

Examples of relationships between agents and artifacts

* Senior management, the subcontract manager, and the software manager review the subcontractor's progress reports, SCM reports, subcontractor SQA report, and SQA's report about the subcontractor's progress.

Key Process Area: Software Configuration Management

Subprocess area: Plan SCM
Action: Plan software configuration management activities

Examples of Agents	*Examples of Artifacts*
SCM management	SCM Plan
Software manager	

Examples of relationships between agents and artifacts

* The software manager, and the SCM management review and accept the SCM plan for the project.

Subprocess area: Create software work product baselines
Action: Identify selected software work products for a baseline, which is controlled and made available

Examples of Agents	*Examples of Artifacts*
Software manager	Baselined work products
Work force	Software Development Plan
Configuration Control Board members	SCM Plan
	Configuration Control Board minutes and agenda

Examples of relationships between agents and artifacts

* The software manager ensures the software plan shows what work products are to be baselined, and that this plan is carried out.
* The Configuration Control Board or equivalent reviews and agrees with all baselining activities.

Subprocess area: Control changes
Action: Control changes to software baselines

Examples of Agents	*Examples of Artifacts*
Software Manager	Trouble reports
Configuration Control Board members	Action items requiring changes to work products
	Configuration Control Board minutes and agenda

Examples of relationships between agents and artifacts

* The Configuration Control Board oversees all activities relating to making changes to work products.

Key Process Area: Software Configuration Management (cont'd)

Subprocess area: Report status
Action: Report software configuration status

Examples of Agents	*Examples of Artifacts*
SCM specialist	SCM reports
Work force	

Examples of relationships between agents and artifacts

- The work force has access to the status of all baselined work products in the care of the SCM engineer.

Key Process Area: Software Quality Assurance

Subprocess area: Plan SQA
Action: Plan software quality assurance activities

Examples of Agents	*Examples of Artifacts*
SQA management	SQA plan
Software manager	

Examples of relationships between agents and artifacts

- The software manager and the SQA management review and accept the SQA plan for the project.

Subprocess area: Perform SQA
Action: Verify adherence of activities and products to applicable standards

Examples of Agents	*Examples of Artifacts*
SQA engineers	Process artifacts (Software Development Files,
Software engineers	project directives, project standards and
Software manager	procedures)
Software supervisors	Product artifacts (Software Requirements
Software testers	Specification, Software Design Document,
	software test cases, software products
	SQA reports

Examples of relationships between agents and artifacts

- SQA engineers verify that the project's process artifacts are being followed and used in the development of the software product.
- SQA engineers verify that the software products are being produced in accordance with the project's process artifacts.

Subprocess area: Communicate results
Action: Communicate SQA results

Examples of Agents	*Examples of Artifacts*
SQA engineers	SQA reports
Work force	SQA engineer entries in software engineering
	process documents
	Software Development Files

Examples of relationships between agents and artifacts

- SQA engineer interaction with the work force is visible and available to all.

Key Process Area: Software Quality Assurance (cont'd)

Subprocess area: Address noncompliance
Action: Address noncompliance issues

Examples of Agents	*Examples of Artifacts*
Senior management	Noncompliance report
Project manager	SQA report
Software manager	

Examples of relationships between agents and artifacts

* Senior management sees that all noncompliance issues are resolved.

Key Process Area: Organizational Process Focus

Subprocess area: Coordinate software process activities
Action: Coordinate software process development and improvement activities

Examples of Agents	*Examples of Artifacts*
Senior management	Training objectives
Project manager	Environment (tools) investigation reports
Software manager	IR&D suggestions
Software supervisors	Overhead expenditure suggestions
Software engineers	Software process improvement plan
Work force	Record of periodic training review
SQA engineers	

Examples of relationships between agents and artifacts

* The managers and work force cooperate to establish continuous process improvement.
* The managers and work force cooperate in identifying changes to the tools within the software development environment.
* The managers and work force cooperate in establishing training curriculum objectives.
* The managers and work force cooperate to identify process improvement investigations for IR&D and overhead programs.

Subprocess area: Assess software processes used
Action: Assess software processes in use against a process standard

Examples of Agents	*Examples of Artifacts*
Senior management	Process standards
Project manager	
Software manager	
Software supervisors	
Software engineers	
Work force	
SQA engineers	

Examples of relationships between agents and artifacts

* The managers and work force use a process standard to identify strengths and weaknesses in the processes used.

Key Process Area: Organizational Process Focus (cont'd)

Subprocess area: Plan software process improvement
Action: Plan software process development and improvement activities

Examples of Agents	*Examples of Artifacts*
Senior management	Software process improvement plan
Project manager	
Software manager	
Software supervisors	
Software engineers	
Work force	
SQA engineers	

Examples of relationships between agents and artifacts

• The managers and work force accept and support the organization's software process improvement plan.

Key Process Area: Organization Process Definition

Subprocess area: Provide standard process
Action: Develop and maintain the organization's standard software process

Examples of Agents	*Examples of Artifacts*
Senior management	Organization process standards
Project manager	
Project support personnel	
Software manager	
Software supervisors	
Software engineers	
Personnel responsible for organization process focus	

Examples of relationships between agents and artifacts

• Software engineering staff develop and maintain software process standards for the organization.

Subprocess area: Retain software process information
Action: Collect, review, and make available the organizational software process data

Examples of Agents	*Examples of Artifacts*
Senior management	Organization process standards
Project manager	
Project support personnel	
Software manager	
Software supervisors	
Software engineers	
Personnel responsible for organization process focus	

Examples of relationships between agents and artifacts

• Software engineering staff review the organization's standards.
• The project staff have available to them the organization's standards.

Key Process Area: Software Product Engineering

Subprocess area: Build software
Action: Build and maintain software according to project's defined software process

Examples of Agents	*Examples of Artifacts*
Software manager	Software Development Plan
Software supervisors	Project directives
Software engineers	Project standards and procedures
SQA engineers	Project training plan (for new methodology and tools)

Examples of relationships between agents and artifacts

- The project software engineers perform their work in accordance with the defined software engineering tasks.
- Project software management explicitly prepares the project work force for the use of new methodologies and tools within their software engineering tasks.

Subprocess area: Ensure consistency
Action: Ensure consistency of software work products

Examples of Agents	*Examples of Artifacts*
Software manager	Traceability matrices
Software supervisors	Test case documentation
Software engineers	
Work force	

Examples of relationships between agents and artifacts

- The work force ensures that the software work products are consistent with each other by complete cross referencing, which includes test cases and traceability matrices.

Key Process Area: Integrated Software Management

Subprocess area: Define project process
Action: Define project's software process by tailoring the organization's standard software process

Examples of Agents	*Examples of Artifacts*
Project manager	Organization standards
Project support personnel	Project directives
Software manager	Project standards and procedures
Software supervisors	
Software engineers	
Personnel responsible for organization process focus	

Examples of relationships between agents and artifacts

- Project staff and software engineering staff tailor the organization standards to create the project's standards and procedures
- The software engineering staff review and accept the project's tailored versions of the organization's standards.

Subprocess area: Manage according to process
Action: Manage project according to its defined process

Examples of Agents	*Examples of Artifacts*
Project manager	Software Development Plan
Project support personnel	Project standards and procedures
Software manager	Project directives
Software supervisors	
Software engineers	
Personnel responsible for organization process focus	
SQA engineers	

Examples of relationships between agents and artifacts

- Project managers, supervisors, and SQA engineers make project plans that are in accordance with the project's standards and procedures.

Key Process Area: Intergroup Coordination

Subprocess area: Agree on customer's requirements
Action: Obtain agreement on the customer's requirements

Examples of Agents	*Examples of Artifacts*
Software managers	Allocated baseline
Software supervisors	Contract documents (e.g., CDRL schedule)
System engineers	Project meeting minutes
Project support personnel	Technical Coordination Meeting minutes
Customer	

Examples of relationships between agents and artifacts

- Work force with direct development responsibility reach a consensus with the customer about the requirements as the requirements mature and evolve.

Subprocess area: Coordinate intergroup commitments
Action: Obtain agreement by affected groups on commitments between engineering groups

Examples of Agents	*Examples of Artifacts*
Software managers	Work authorization sheet (descriptions of work)
Software supervisors	Meeting minutes
System engineers	Action items
Project support personnel	
Work force groups	
Customer	

Examples of relationships between agents and artifacts

- Work force groups are aligned and working to detailed plans that are integrated and coordinated.

Subprocess area: Manage intergroup issues
Action: Identify, track, and resolve intergroup issues

Examples of Agents	*Examples of Artifacts*
Software managers	Action items
Software supervisors	Software trouble reports
System engineers	Project meeting minutes
Project support personnel	
Work force groups	
Customer	

Examples of relationships between agents and artifacts

- Work force groups work together to resolve intergroup issues.

Key Process Area: Peer Reviews

Subprocess area: Plan peer reviews
Action: Plan peer review activities

Examples of Agents	*Examples of Artifacts*
Software engineers	Peer Review agendas
Software testers	Software Development Files
SQA engineers	SQA reports

Examples of relationships between agents and artifacts

* The software engineers plan peer reviews.

Subprocess area: Identify and remove defects
Action: Identify and remove defects in the software work products

Examples of Agents	*Examples of Artifacts*
Software engineers	Peer Review reports
Software testers	
System engineers	
SQA engineers	

Examples of relationships between agents and artifacts

* All stakeholders are involved in removing defects found during peer reviews.

Key Process Area: Training Program

Subprocess area: Plan training
Action: Plan training activities

Examples of Agents	*Examples of Artifacts*
Training staff	Training requirements for each work force position
	Training plan (includes schedule)
	Training curriculum
	Records of planning meeting on required training

Examples of relationships between agents and artifacts

- Training staff plans the training plan and curriculum.

Subprocess area: Provide training
Action: Provide training

Examples of Agents	*Examples of Artifacts*
Training staff	Training requirements for each work force position
Work force	Training plan (includes schedule)
	Training curriculum

Examples of relationships between agents and artifacts

- Training staff provide the training required by the organization's work force.

Subprocess area: Receive necessary training
Action: Receive necessary training

Examples of Agents	*Examples of Artifacts*
Work force	Training attendance records
	Training plan

Examples of relationships between agents and artifacts

- Individuals in the organization receive training to carry out their roles.

Key Process Area: Quantitative Process Management

Subprocess area: Plan QPM
Action: Plan quantitative process management for the project

Examples of Agents	*Examples of Artifacts*
Process staff	QPM plan
Software engineers	
Software testers	
System engineers	
SQA engineers	

Examples of relationships between agents and artifacts

• All stakeholders in a process are aware of the process goals.

Subprocess area: Control process quantitatively
Action: Control project's process quantitatively

Examples of Agents	*Examples of Artifacts*
Software engineers	QPM plans
Software testers	Process goal measurement data and profiles
System engineers	
SQA engineers	

Examples of relationships between agents and artifacts

• All users of a process analyze their process performance against the established limits.

Subprocess area: Establish organization's process capability
Action: Analyze and combine process performance of an organization's projects

Examples of Agents	*Examples of Artifacts*
Process staff	Process goal measurement data and profiles
	Process capability reports

Examples of relationships between agents and artifacts

• Process staff analyze the process goal measurements to derive capability reports.

Key Process Area: Software Quality Management

Subprocess area: Plan quality management
Action: Plan software quality management

Examples of Agents	Examples of Artifacts
Process staff	SQM plan
Project staff	

Examples of relationships between agents and artifacts

- Process staff analyzes customer and end user needs to define an SQM plan.

Subprocess area: Define software quality products
Action: Define measurable, prioritized quality goals

Examples of Agents	Examples of Artifacts
Process staff	Project quality goal documentation
Project staff	

Examples of relationships between agents and artifacts

- Project staff analyzes the organization's needs and priorities and the customer and end user needs to assign priorities to the products quality goals for products.

Subprocess area: Track quality progress
Action: Track progress toward achieving quality goals

Examples of Agents	Examples of Artifacts
Process staff	Project quality goal documentation
Project staff	Action items about the project's quality goals for a project

Examples of relationships between agents and artifacts

- Project staff analyzes the organization's needs and priorities and the customer and end user needs to assign priorities to the products quality goals for products.

Key Process Area: Defect Prevention

Subprocess area: Plan defect prevention
Action: Plan defect prevention activities

Examples of Agents	*Examples of Artifacts*
Process staff	Defect prevention plan
Project staff	

Examples of relationships between agents and artifacts

* Project and process staff identify the defect prevention activities for the plan.

Subprocess area: Identify defect causes
Action: Identify common causes of defects

Examples of Agents	*Examples of Artifacts*
Process staff	Defect prevention plan
Project staff	Defect records and trend reports
	Action items identifying common causes

Examples of relationships between agents and artifacts

* Project and process staff identify the defect prevention activities for the plan.

Subprocess area: Eliminate defect causes
Action: Prioritize and eliminate causes of defects

Examples of Agents	*Examples of Artifacts*
Process staff	Defect records and trends
Project staff	Proposals to remove common causes

Examples of relationships between agents and artifacts

* Project and process staff analyze common causes of action plans and develop proposals for their elimination.

Key Process Area: Technology Change Management

Subprocess area: Plan technology changes
Action: Plan incorporation of technology changes

Examples of Agents	*Examples of Artifacts*
Process staff	Technology change plan
Project staff	Performance records of the organization's standard software process
	Procedures for installing new technology

Examples of relationships between agents and artifacts

- Project and process staff analyze the relationship between new technological possibilities with the organization's standard process performance record.

Subprocess area: Evaluate new technologies
Action: Determine effect of new technologies on quality and productivity

Examples of Agents	*Examples of Artifacts*
Process staff	Technology change evaluation results
Project staff	Piloting results
	Explicit goals for quality and productivity improvements

Examples of relationships between agents and artifacts

- Project and process staff establish productivity and quality improvement goals.

Subprocess area: Adopt new technology
Action: Transfer appropriate new technologies into practice

Examples of Agents	*Examples of Artifacts*
Process staff	Transfer plans
Project staff	Updated versions of the organization's standards and procedures
Work force	Training curriculum
	Training schedule and attendance

Examples of relationships between agents and artifacts

- Project and process staff establish productivity and quality improvement goals.

Key Process Area: Process Change Management

Subprocess area: Plan process improvement
Action: Plan continuous process improvement

Examples of Agents	*Examples of Artifacts*
Senior management	Process change plan
Process staff	Organization policy for continuous process
Project staff	improvement
Work force	Process change procedure

Examples of relationships between agents and artifacts

- The work force, project staff, and process staff are involved in the decision making about planning continuous process improvement.

Subprocess area: Empower everyone
Action: Empower organization people to participate in process improvement

Examples of Agents	*Examples of Artifacts*
Senior management	Process improvement review agendas and meeting
Process staff	minutes
Project staff	Comment sheets
Work force	Procedure for settling differences about comments

Examples of relationships between agents and artifacts

- The work force comments about continuous process improvement activities are dealt with according to a procedure that has wide acceptance.

Subprocess area: Continuously improve
Action: Continuously identify and manage process improvement implementations

Examples of Agents	*Examples of Artifacts*
Senior management	Process improvement implementation review
Process staff	agendas and meeting minutes
Project staff	Implementation plans
Work force	Training curriculum
	Training schedule and attendance

Examples of relationships between agents and artifacts

- The work force comments about continuous process improvement activities are dealt with according to a procedure that has wide acceptance.

Section G.2   Standard Look-for Table

How These Tables Should Be Used

The Standard Look-for Table helps SCE teams determine whether the goal of a KPA has been satisfied with respect to each of the common features. This table is applicable to all subprocess areas; specific guidance that applies to the activities that are unique for a particular subprocess area is found in the next section, "Probing Guides for Key Process Areas" (➡page G-36).

The guidelines in this table help the teams evaluate the software processes the organization has implemented. They contain information about characteristics that any process should have to be effective, repeatable, and lasting regardless of the specific activities performed. These guidelines may be used both to help focus the investigation and to help the team make judgements about the *information collected*.

The table contains probing guides that indicate what the SCE team might investigate relative to each of the common features. The probing guides are "generic" statements about what to look for relative to software processes, supported by subordinate phrases which provide additional meaning. The subordinate phrases are derived from the CMM v1.1 key practices. The phrases are extractions of the major ideas in the key practices; the table explicitly references relevant key practices [Paulk 93b].

Table Format

The table is split across facing pages. The left most column contains the probing guides. The rest of the table has columns for each KPA (listed across the top), with annotations in the columns for specific key practices that are closely related to the text in the probing guides.

Example

For topic refinement, the team can use the probing guides as a starting point. For example, suppose that the Software Project Planning KPA is being investigated. The subprocess area "develop estimates" is on the Critical Subprocess Area List.

Examining the leftmost column, the first probing guide is "evidence exists that the organization is committed to KPA goals", and possible topics include checking that projects follow organizational policies. Further down we see "evidence exists that the organization has the ability to meet KPA goals", with possible topics relating to resources, training, etc. All of these considerations could apply to the "develop estimates" subprocess area. Consideration of the agents, artifacts, and relationships will yield interview topics and potential documents for review. See "Agent, Artifact, Relationship Examples" (➡page G-3).

To make judgments, the team would examine all of the evidence collected relative to the organization's estimation process, and then ask the question "does sufficient evidence exist that the organization has the ability to meet the goals of this KPA?" Each of the areas listed might be a factor the team would consider when making their decision.

Standard Look For Table

Key Process Areas

Probing Guides	Requirements Management	Software Project Planning	Software Project Tracking and Oversight	Software Subcontract Management	Software Quality Assurance	Software Configuration Management	Organization Process Focus
Evidence exists that the organization is committed to KPA goals:							
Senior Management sponsors the implementation of organizational policies							C2
Leadership is established for implementation of organizational policies		C1	C1	C2			C3
Projects follow organizational policies in implementing KPA practices	C1	C2	C2	C1	Ab1	C1	C1
Evidence exists that the organization has the ability to meet KPA goals:							
Roles and responsibilities for implementation of KPA practices are defined	Ab1	Ab2	Ab2		Ab1	Ab1 Ab2	Ab1
Adequate resources are provided for implementation of KPA practices	Ab3	Ab3	Ab3	Ab1	Ab2	Ab3	Ab2
Adequate training is provided for the individuals responsible for implementation of KPA practices	Ab4	Ab4	Ab4	Ab2	Ab3	Ab4	Ab3
Adequate orientation is provided for individuals affected by implementation of KPA practices			Ab5	Ab3	Ab4	Ab5	Ab4

Organization Process Definition	Training Program	Integrated Software Management	Software Product Engineering	Intergroup Coordination	Peer Reviews	Quantitative Process Management	Software Quality Management	Defect Prevention	Technology Change Management	Process Change Management
									C2	C2
									C3	
C1	C1	C1	C1	C1	C1	C1 C2	C1	C1	C1	C1
	Ab1					Ab1		Ab1 Ab2	Ab1	
Ab1	Ab2	Ab1	Ab1	Ab1 Ab2	Ab1	Ab2 Ab3	Ab1	Ab3	Ab2 Ab3 Ab4	Ab1
Ab2	Ab3	Ab2 Ab3	Ab2	Ab3	Ab2 Ab3	Ab3	Ab2	Ab4	Ab5	Ab2 Ab3 Ab4
	Ab4		Ab3 Ab4	Ab4 Ab5		Ab4	Ab3			

Standard Look For Table (cont'd)

Key Process Areas

Probing Guides	Requirements Management	Software Project Planning	Software Project Tracking and Oversight	Software Subcontract Management	Software Quality Assurance	Software Configuration Management	Organization Process Focus
Evidence exists that the activities critical to KPA goals are defined:							
Documented standards and procedures exist for implementation of KPA practices		Ac4 Ac6 Ac9 Ac10 Ac11 Ac12	Ac2 Ac3 Ac13	Ac1 Ac2 Ac6 Ac9 Ac10 Ac11 Ac12	Ac1 Ac7	Ac1 Ac5 Ac6 Ac7 Ac8 Ac10	
Evidence exists that performance of KPA activities is measured and analyzed:							
Measurements are made and used to determine the status of activities	M1	M1		M1	M1	M1	M1
Evidence exists that the organization verifies the implementation KPA practices:							
Senior Management reviews activities on a periodic basis	V1	V1	V1	V1	V1	V1	V1
Project Management reviews activities on a periodic and event driven basis	V2	V2	V2	V2	V2	V2	
An independent group reviews and audits the implementation of KPA practices and reports results.	V3	V3	V3	V3	V3	V3	V3

Organization Process Definition	Training Program	Integrated Software Management	Software Product Engineering	Intergroup Coordination	Peer Reviews	Quantitative Process Management	Software Quality Management	Defect Prevention	Technology Change Management	Process Change Management
Ac1	Ac2	Ac2		Ac4	Ac2	Ac1	Ac1	Ac3	Ac5	Ac3
Ac2		Ac3		Ac6		Ac4		Ac6	Ac7	Ac5
		Ac6				Ac5		Ac7	Ac8	Ac8
		Ac7				Ac7				
		Ac8								
		Ac9								
		Ac10								
	M1	M1	M1	M1	M1	M1	M1	M1	M1	M1
	M2		M2							
	V1	V1	V1	V1		V1		V1	V1	V1
	V2	V2	V2			V2	V2	V2		
V3	V2	V3	V3	V3	V1	V3	V3	V3	V2	V2
	V3									

Section G.3   Probing Guides for Key Process Areas

How These Tables Should Be Used

The probing guides help the teams to determine if the KPA goals have been satisfied. They provide information about the types of activities associated with each subprocess area. These guidelines can be used to help focus the investigation for specific subprocess areas, and also to help the team make judgements about the data collected.

In the previous section, "Standard Look-for Table" (⟿page G-29), the tables pertained to all subprocess areas. The tables in this section supplement the information in the Standard Look-for Table by providing information related to the specific activities associated with a specific subprocess area.

For topic refinement, the probing guides provide examples of specific types of activities that might be found.

When making judgments, the probing guides are indications of what an SCE team may be able to observe. They are necessary for goal achievement but not sufficient. The team must evaluate all of the data collected against the KPA goal statement. Teams generate findings by testing the information collected against the goal statement to determine if the goal is satisfied.

Table Format

The Probing Guides are organized by KPA and subprocess area. There is one table for each subprocess area in the KPA.

Each table contains the KPA goal statement which corresponds to the subprocess area. The goals specified in the CMM are used.

Each goal is associated with one or more probing guides.

Probing guides are supported by subordinate phrases which help to add meaning to the probing guides. The subordinate phrases are derived from the CMM key practices, and are extractions of the major ideas in the key practices.

Each subordinate phrase is followed by a reference to the CMM key practices from which it was derived. The format for the reference is an abbreviation to indicate the type of key practice followed by   nber.

Key Process Area: Requirements Management

Subprocess Area	Establish and maintain requirements baseline
Goal	*System requirements allocated to software are controlled to establish a baseline for software engineering and management use (G1).*
Probing Guides	Evidence exists that the allocated baseline is managed: • Project responsibility for requirements allocation is established (Ab1), • Allocated requirements are documented (Ab2), • The software engineering group reviews the allocated requirements (Ac1), • The allocated requirements are the basis for software plans, work products, and activities (Ac2).

Subprocess Area	Manage requirements-driven changes
Goal	Software plans, products, and activities are kept consistent with the system requirements allocated to software (G2).
Probing Guides	Evidence exists that consistency is maintained between the allocated requirements and software plans, products, and activities: • Changes to allocated requirements are reviewed for impact (Ac3), • The project is replanned and software products are modified, as necessary, to reflect changes to allocated requirements (Ac3).

Key Process Area: Software Project Planning

Subprocess Area	Develop estimates
Goal	Software estimates are documented for use in planning and tracking the software project (G1).
Probing Guides	Evidence exists that there is a defined process for deriving and recording the estimates used in software planning: • Software product size, cost/effort, critical computer resource, and schedule estimates are derived (Ac9, Ac10, Ac11, Ac12). • Risks associated with estimates are identified (Ac13). • This software planning data is recorded (Ac15).

Subprocess Area	Plan software activities
Goal	Software project activities and commitments are planned and documented (G2).
Probing Guides	Evidence exists that software activities are planned and documented: • Software project planning is initiated early in overall project planning (Ac2). • Plans are based on an approved and documented statement of work (Ab1). • Plans are based on an identified software lifecycle (Ac5). • Plans address estimates (Ac9, Ac10, Ac11, Ac12). • Plans are documented (Ac6, Ac7). • Plans identify software work products (Ac8). • Plans include facilities and support tools (Ac14).

Subprocess Area	Make commitments
Goal	Affected groups and individuals agree to their commitments related to the software project (G3).
Probing Guides	Evidence exists that affected groups agree to the plans which affect them: • The software engineering group participates on the proposal team (Ac1). • The software engineering group participates in project planning throughout the project life (Ac3). • Senior management reviews external commitments (Ac4).

Key Process Area: Software Project Tracking and Oversight

Subprocess Area	Track progress
Goal	Actual results and performances are tracked against the software plans (G1).
Probing Guides	Evidence exists that actual results and performance are tracked against documented plans: • An approved, documented software development plan is used for tracking (Ab1, Ac1), • Software product size, effort/cost, critical computer resources, and schedule estimates are tracked (Ac5, Ac6, Ac7, Ac8), • Technical activities are tracked (Ac9), • Risks are tracked (Ac10), • Measurement and replanning data are recorded (Ac11).

Subprocess Area	Take corrective action
Goal	Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans (G2).
Probing Guides	Evidence exists that significant deviations from plans leads to initiation of corrective actions: • Corrective actions are taken to address significant variance is product size, effort/cost, critical computer resources, and schedule variances (Ac5, Ac6, Ac7, Ac8), • Corrective actions are taken to address technical issues (Ac9). Evidence exists that corrective actions are managed to closure: • Internal reviews are held to track progress, plans, performance, and issues (Ac12), • Formal reviews are held to address results (Ac13).

Key Process Area: Software Project Tracking and Oversight (cont'd)

Subprocess Area	Manage commitment changes
Goal	Changes to software commitments are agreed to by the affected groups and individuals (G3).
Probing Guides	Evidence exists that explicit agreements are made for changes in commitments: • The process for revising the software development plan [which documents commitments] is defined (Ac2), • Senior management reviews changes to external commitments (Ac3), • Affected groups are informed of changes to commitments (Ac4).

Key Process Area: Software Subcontract Management

Subprocess Area	Select subcontractors
Goal	The prime contractor selects qualified software subcontractors (G1).
Probing Guides	Evidence exists that qualified software subcontractors are selected: • The work to be subcontracted is defined and planned (Ac1), • Subcontractors are selected based on their ability to perform the work (Ac2).

Subprocess Area	Establish and maintain commitments
Goal	The prime contractor and the software subcontractor agree to their commitments to each other (G2).
Probing Guides	Evidence exists that commitments the prime contractor and software subcontractor agree to their mutual commitments: • Subcontract management is based on the subcontract contractual agreement (Ac3). • The subcontractor's software development plan is reviewed and approved (Ac4). • Changes to commitments are managed by defined procedures (Ac6).

Subprocess Area	Maintain communications
Goal	The prime contractor and the software subcontractor maintain ongoing communications (G3).
Probing Guides	Evidence exists that interaction with subcontractor is actively maintained throughout development: • Status/coordination reviews are held (Ac7). • Technical reviews and interchanges are held (Ac8). • Formal reviews are held at selected milestones (Ac9). • Subcontractor performance is reviewed with the subcontractor (Ac13).

Key Process Area: Software Subcontract Management (cont'd)

Subprocess Area	Track progress
Goal	The prime contractor tracks the software subcontractor's actual results and performance against its commitments (G4).
Probing Guides	Evidence exists that the subcontractor's progress is actively monitored: • Subcontract activities are tracked against the plan (Ac5), • Formal reviews are held at selected milestones (Ac9), • Software quality assurance activities are monitored (Ac10), • Software configuration management activities are monitored (Ac11), • Acceptance testing of subcontract software products is performed (Ac12), • Subcontractor performance is evaluated (Ac13).

Key Process Area: Software Quality Assurance

Subprocess Area	Plan SQA
Goal	Software quality assurance activities are planned (G1).
Probing Guides	Evidence exists that SQA activities are planned: • An SQA plan is prepared for the project (Ac1), • SQA activities are performed according to plan (Ac2).

Subprocess Area	Perform SQA
Goal	Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively (G2).
Probing Guides	Evidence exists that adherence of software products and activities to standards, procedures and requirements is verified objectively: • SQA reviews software development plans, standards, and procedures (Ac3), • SQA verifies compliance of software engineering activities with plans, standards, and procedures (Ac4), • SQA verifies compliance of work products to standards (Ac5).

Subprocess Area	Communicate results
Goal	Affected groups and individuals are informed of software quality assurance activities and results (G3).
Probing Guides	Evidence exists that there is active communication of SQA activities: • SQA periodically reports the results of its activities to the software engineering group (Ac6), • SQA periodically reviews its activities with customer's SQA personnel (Ac8).

Key Process Area: Software Quality Assurance (cont'd)

Subprocess Area	Address noncompliance
Goal	Noncompliance issues that cannot be resolved within the software project are addressed by senior management (G4).
Probing Guides	Evidence exists that senior management is informed of unresolved noncompliance issues: • The SQA group has an independent reporting chain to senior management (C1) • Deviations identified in SQA reviews and audits are documented and handled according to a defined procedure (Ac7). • Noncompliance items not resolved within the project are presented to senior management for resolution (Ac7).

Key Process Area: Software Configuration Management

Subprocess Area	Plan SCM
Goal	Software configuration management activities are planned (G1).
Probing Guides	Evidence exists that SCM activities are planned: • A project SCM plan is documented and approved (Ac1), • The SCM plan is the basis for performing SCM activities (Ac2).

Subprocess Area	Create software work products baselines
Goal	Selected software work products are identified, controlled, and available (G2).
Probing Guides	Evidence exists that selected software work products are baselined and controlled: • A configuration management library is established as a repository for software baselines (Ac3), • Software work projects which are to be placed under configuration management are identified (Ac4), • Creation and release of baseline products from baseline repositories is controlled (Ac7).

Subprocess Area	Control changes
Goal	Changes to identified software work products are controlled (G3).
Probing Guides	Evidence exists that changes to all baselined products are controlled: • The authority for managing the project's software baseline, such as a Configuration Control Board, is established (Ab1), • Configuration item change requests and problem reports are tracked (Ac5), • Changes to baselines are controlled (Ac6), • Baseline audits are conducted. (Ac10, V3).

Key Process Area: Software Configuration Management (cont'd)

Subprocess Area	Report status
Goal	Affected groups and individuals are informed of the status and content of software baselines (G4).
Probing Guides	Evidence exists that baseline status and content is communicated to affected groups: • The status of configuration items is recorded (Ac8), • Reports documenting SCM activities and the contents of the baseline are distributed to affected groups (Ac9).

Key Process Area: Organization Process Focus

Subprocess Area	Coordinate software process activities
Goal	Software process development and improvement activities are coordinated across the organization (G1).
Probing Guides	Evidence exists that software process development and improvement is coordinated organization wide: • Senior management sponsors process development and improvement activities (C2), • Senior management oversees process development and improvement activities (C3), • Development and improvement activities are coordinated at the organization level (Ac3), • Use of the organizational level software data base is coordinated (Ac4), • New processes, methods, and tools are transferred as appropriate (Ac5), • Training in software processes is coordinated (Ac6), • Process development and improvement activities are communicated (Ac7).

Subprocess Area	Assess software processes used
Goal	The strengths and weaknesses of the software processes used are identified relative to a process standard (G2).
Probing Guides	Evidence exists that the strengths and weaknesses of software processes are identified: • The software process is assessed periodically (Ac1).

Subprocess Area	Plan SPI
Goal	Organization-level process development and improvement activities are planned (G3).
Probing Guides	Evidence exists that software process development and improvement activities are planned: • A group responsible for the organization's software process exists (Ab1), • Action plans are developed to address assessment findings (Ac1), • The organization maintains a software process development and improvement plan based on the action plans (Ac2).

Key Process Area: Organization Process Definition

Subprocess Area	Provide standard process
Goal	A standard software process for the organization is developed and maintained (G1).
Probing Guides	Evidence exists that a standard software process is defined: • The standard software process is developed and maintained (Ac1), • The standard software process is documented (Ac2), • Standard development life cycles are documented and approved for use (Ac3), • Guidelines and criteria for tailoring the standard software process exist for project use (Ac4).

Subprocess Area	Retain software process information
Goal	Information related to the use of the organization's standard software process by the software projects is collected, reviewed, and made available (G2).
Probing Guides	Evidence exists that information concerning standard process use is monitored and available: • An organization software process database is established and maintained (Ac5), • A library of software process-related documentation is established and maintained (Ac6).

Key Process Area: Training Program

Subprocess Area	Plan training
Goal	Training activities are planned (G1).
Probing Guides	Evidence exists that training plans reflect needs: • A group responsible for fulling the organization's training needs exists (Ab1), • Each project develops and maintains a training plan which specifies its training needs (Ac1), • An organizational training plan is developed and maintained (Ac2).

Subprocess Area	Provide training
Goal	Training for developing the skills and knowledge needed to perform software management and technical roles is provided (G2).
Probing Guides	Evidence exists that necessary training is provided: • Training is provided in accordance with the organizational training plan (Ac3), • Training courses are prepared to organization standards (Ac4).

Subprocess Area	Receive necessary training
Goal	Individuals in the software engineering group and software-related groups receive the training necessary to perform their roles (G3).
Probing Guides	Evidence exists that individuals are trained to perform their roles: • Individuals are trained unless they possess the knowledge and skills required to perform their role (Ac3), • A waiver procedure for required training is established and used (Ac5), • Training records are maintained (Ac6).

Key Process Area Integrated Software Management

Subprocess Area	Define project process
Goal	The project's defined software process is a tailored version of the organization's standard software process (G1)
Probing Guides	Evidence exists that project process is a tailored version of the organization process: • The project's software process is developed by tailoring the organization's standard software process (Ac1), • The project's software process is revised according to a defined procedure (Ac2), • The software development plan describes process use (Ac3).

Subprocess Area	Manage according to process
Goal	The project is planned and managed according to the project's defined software process (G2).
Probing Guides	Evidence exists that project is planned and managed according to the project's defined process: • The software project is managed according to the project's defined process (Ac4), • The organization's software data base is used for software planning and estimating (Ac5), • The project manages the software product size, effort/cost, critical computer resources, schedule, and risks according to a defined procedure (Ac6, Ac7, Ac8, Ac9, Ac10), • Reviews of software project performance are periodically performed (Ac11).

Key Process Area Software Product Engineering	
Subprocess Area	**Build software**
Goal	The software engineering tasks are defined, integrated, and consistently performed to produce the software (G1).
Probing Guides	Evidence exists that software engineering tasks are performed consistently: • Software engineering methods and tools are integrated into the project's defined software process (Ac1), • Software requirements are analyzed, software is designed, coded, tested, and integrated, and operations and maintenance documentation is developed according to the project's defined software process (Ac2, Ac3, Ac4, Ac5, Ac6, Ac8), • Data on defects is collected and analyzed according to the projects defined software process (Ac9).
Subprocess Area	**Ensure consistency**
Goal	Software work products are kept consistent with each other (G2).
Probing Guides	Evidence exists that work products are consistent: • Consistency is maintained across software work products (Ac10), • System and acceptance testing demonstrate adherence to requirements (Ac7).

Key Process Area: Intergroup Coordination

Subprocess Area	Agree on customer's requirements
Goal	The customer's requirements are agreed to by all affected groups (G1).
Probing Guides	Evidence exists that there is agreement on customer requirements: • Engineering groups (including software) participate with customers in establishing requirements (Ac1).

Subprocess Area:	Coordinate intergroup commitments
Goal	The commitments between the engineering groups are agreed to by the affected groups (G2).
Probing Guides	Evidence exists that affected groups agree on commitments: • A documented plan communicates commitments (Ac3), • Critical dependencies are identified and negotiated (Ac4).

Subprocess Area	Manage intergroup issues
Goal	The engineering groups identify, track, and resolve intergroup issues (G3).
Probing Guides	Evidence exists that intergroup issues are tracked and resolved: • Representatives of engineering groups (including software) work together to coordinate technical activities and resolve technical issues (Ac2), • A documented plan is used to coordinate and track work performed (Ac3), • Critical dependencies are managed (Ac4), • Work products are reviewed by receiving groups (Ac5), • Unresolved issues are handled according to a defined procedure (Ac6), • Periodic technical reviews and interchanges are held (Ac7).

Key Process Area: Peer Reviews

Subprocess Area	Plan peer reviews
Goal	Peer review activities are planned (G1).
Probing Guides	Evidence exists that peer reviews are planned: • Peer review plans are documented (Ac1).

Subprocess Area	Identify and remove defects
Goal	Defects in the software work products are identified and removed (G2).
Probing Guides	Evidence exists that software product defects are identified and removed: • Peer reviews are conducted according to a defined procedure (Ac2). • Results of peer reviews are recorded (Ac3).

Key Process Area: Quantitative Process Management

Subprocess Area	Plan QPM
Goal	The quantitative process management activities are planned (G1).
Probing Guides:	Evidence exists that quantitative process management activities are planned: • A group responsible for coordinating organization QPM activities exists (Ab1), • A project plan for QPM is developed (Ac1), • QPM activities are performed according to plan (Ac2), • The strategy for QPM data collection and analysis is based on the project's defined software process (Ac3).

Subprocess Area	Control process quantitatively
Goal	The process performance of the project's defined software process is controlled quantitatively (G2).
Probing Guides	Evidence exists that the software process is controlled quantitatively: • QPM data is collected (Ac4), • QPM data is used to analyze and control the project software process (Ac5).

Subprocess Area	Establish organization's process capability
Goal	The process capability of the organization's standard software process is known in quantitative terms (G3).
Probing Guides	Evidence exists that process capability in known in quantitative terms: • Reports documenting QPM results are prepared and distributed (Ac6), • A process capability baseline for the organization's standard process is established and maintained (Ac7).

Key Process Area: Software Quality Management

Subprocess Area	**Plan quality management**
Goal	The project's software quality management activities are planned (G1).
Probing Guides	Evidence exists that software quality management activities are planned:

- A software quality plan is developed and maintained (Ac1).
- The project's software quality management activities are based on the software quality plan (Ac2).

Subprocess Area	**Define software quality goals**
Goal	Measurable goals for software product quality and their priorities are defined (G2).
Probing Guides	Evidence exists that measurable goals are defined for product quality:

- Quantitative quality goals for software products are defined, monitored, and revised throughout the software life cycle (Ac3),
- Quality goals are allocated to subcontractor products (Ac5).

Subprocess Area	**Track quality progress**
Goal	Actual progress toward achieving the quality goals for the software products is quantified and managed (G3).
Probing Guides	Evidence exists that actual progress toward achieving product quality goals is quantified and managed:

- Software product quality is measured, analyzed, and compared to goals (Ac4).

Key Process Area: Defect Prevention

Subprocess Area	Plan defect prevention
Goal	Defect prevention activities are planned (G1).
Probing Guides	Evidence exists that defect prevention activities are planned: • Teams are established to coordinate defect prevention activities at the project and organizational level (Ab1, Ab2), • A project plan for defect prevention activities is developed and maintained (Ac1), • Task activities and related defect prevention activities are coordinated by the team performing the task prior to its start (Ac2).

Subprocess Area	Identify defect causes
Goal	Common causes of defects are sought out and identified (G2).
Probing Guide	Evidence exists that common causes of defects are identified: • Causal analysis meetings are conducted (Ac3), • Periodic reviews are held to coordinate implementation of action proposals (Ac4).

Subprocess Area	Eliminate defect causes
Goal	Common causes of defects are prioritized and systematically eliminated (G3).
Probing Guides	Evidence exists that common causes of defects are removed: • Defect prevention data are documented and tracked (Ac5), • Revisions resulting from defect prevention actions are incorporated into the organization's standard software process (Ac6), • Revisions resulting from defect prevention actions are incorporated into the project's standard software process (Ac7), • Software engineering and software-related groups receive periodic feedback on the status and results of defect prevention activities (Ac8).

Key Process Area: Technology Change Management

Subprocess Area	Plan technology changes
Goal	Incorporation of technology changes are planned (G1).
Probing Guides	Evidence exists that technology changes are planned: • Senior management sponsors the organization's activities for change management (C2) • Senior management oversees the organizations's technology change management activities (C3) • A group responsible for technology change management exists (Ab1), • The organization develops and maintains a plan for technology change management (Ac1).

Subprocess Area	Evaluate new technologies
Goal	New technologies are evaluated to determine their effect on quality and productivity (G2).
Probing Guides	Evidence exists that new technologies are evaluated: • Appropriate data on software processes and work products to support evaluation are available (Ab4), • Software projects participate in identifying areas of new technology (Ac2), • The organization's standard software process is systematically analyzed to identify areas that would benefit from new technology (Ac4), • The process for selection and acquisition of new technology is defined (Ac5), • Pilot efforts are conducted, where appropriate, before a new technology is introduced into normal practice (Ac6).

Subprocess Area	Adopt new technology
Goal	Appropriate new technologies are transferred into normal practice across the organization (G3).
Probing Guides	Evidence exists that new technologies are transferred: • Projects are informed of new technologies (Ac3), • Procedures for incorporating new technology into the organization's standard process are defined (Ac7), • Procedures for incorporating new technology into a project's standard process are defined (Ac8).

Key Process Area: Process Change Management

Subprocess Area	Plan process improvement
Goal	Continuous process improvement is planned (G1).
Probing Guides	Evidence exists that continuous process improvement is planned:

• Senior Management Sponsors the organization's activities for software process improvement (C2)
• The group responsible for the organization's software process activities coordinates the software process improvement activities (Ac2),
• The organization develops and maintains a software process improvement plan (Ac3),
• Software process improvement activities are performed in accordance with the software process improvement plan (Ac4) |

Subprocess Area	Empower everyone
Goal	Participation in the organization's software process improvement activities is organization wide (G2).
Probing Guides	Evidence exists that process improvement participation is organization wide:

• A process improvement program is establish which empowers members of the organization (Ac1),
• Organization members participate in the development of software process improvements (Ac6),
• Staff receives feedback on the status and results of software process improvement activities (Ac10). |

Subprocess Area	Continuously improve
Goal	The organization's standard software process and the projects' defined software processes are improved continuously (G3).
Probing Guides	Evidence exists that defined processes are continuously improved:

• Software process improvement activities are performed in accordance with the software process improvement plan (Ac4)
• Handling of software process improvement proposals are defined (Ac5),
• Software process improvements, where appropriate, are installed on a pilot basis prior to their introduction into normal practice (Ac7),
• The procedure for deciding to transfer a software process improvement into normal practice is defined (Ac8),
• Records of software process improvement activities are maintained (Ac9).
• Staff receives feedback on the status and results of software process improvement activities (Ac10). |

Appendix H   Checklists

This appendix contains checklists to serve as reminders of things that need to be done when planning and preparing for an SCE site visit.

Section H.1   Preparing to Use SCE for a Specific Application (⮕page 4-2)

The following checklists may be used by team members to make sure that the necessary preparations have been made by the sponsoring agency. The first checklist is for SCE used in a source selection and the second is for SCE used in contract monitoring.

Checklist for Preparing to Use SCE in a Source Selection

❏ Acquisition Plan includes guidance to determine if SCE should be performed.

❏ Acquisition Announcements include a statement that an SCE will be conducted.

❏ Source Selection plan describes how the results of the SCE will be applied and integrated into the source selection process.

❏ Pre-proposal conference includes a briefing on what to expect when an SCE is conducted.

❏ SCE is incorporated in the evaluation plan.

❏ The Request for Proposals delineates SCE as a criterion and includes instructions for supplying the information required to prepare for an SCE.

❏ Proposals are received and evaluation is completed on schedule.

❏ SCE Information to be provided by the offerors has been received by the team.

❑ The sponsoring agency has determined how frequently the SCEs will be conducted and how the results will be used

❑ The development organization has been briefed on SCE

❑ The sponsoring agency and development organization have agreed to and documented a plan for software process improvement

❑ The contract or contract modification specifies how SCEs will be conducted and how the results will be used

❑ The information that the development organization must supply has been requested

❑ Information to be provided by the development organization has been received

Section H.2   Entry Criteria for Each Phase of the SCE Method

The following checklists may be used at the start of each new phase of the SCE method to verify that the necessary preparations for that phase have been completed.

Entry Criteria for
Phase 1: Evaluation
Start (➡page 4-14)

❑ The decision has been made to use the SCE method.
❑ The product to be built has been defined.

Entry Criteria for
Phase 2: General
Preparation
(➡page 5-2)

❑ The steps of Phase 1 have been completed. The following outputs of Phase 1 are used in this phase:	page 4-14
❑ Target Product Profile	page 4-15
❑ Target Process Capability	page 4-16
❑ The SCE team has received the information requested from the development organizations. The following information is used in this phase:	page 4-7
❑ Proposed Project Profiles	page 4-8
❑ Project Profiles	page 4-9
❑ Organization Charts	page 4-10

Entry Criteria for
Phase 3: Specific
Preparation
(➥page 5-29)

❏ The steps of Phase 1 have been completed. The following outputs of Phase 1 are used in this phase:	page 4-14
❏ Target Product Profile	page 4-15
❏ Target Process Capability	page 4-16
❏ The steps of Phase 2 have been completed. The following outputs of Phase 2 are used in this phase:	page 5-2
❏ Mismatch Identification Tables	page 5-9
❏ Key Issue Table	page 5-23
❏ Validation Worksheets	page 5-27
❏ The SCE team has received the information requested from the development organizations. The following information is used in this phase:	page 4-7
❏ Proposed Project Profiles	page 4-8
❏ Project Profiles	page 4-9
❏ Organization Charts	page 4-10
❏ Responses to the Maturity Questionnaire (if used)	page 4-11

Entry Criteria for
Phase 4: Site Data
Collection (➡page
6-11)

❏ The steps of Phase 3 have been completed. The following outputs of Phase 3 are used in this phase:	page 5-29
❏ Validation Worksheets	page 5-41
❏ Interview Worksheets	page 5-44
❏ Interview Schedule	page 5-44
❏ Entry Briefing	page 5-52
❏ The site visit has been arranged	
❏ The development organization has designated a site visit coordinator	page 6-6
❏ The site visit has been scheduled	page 6-3
❏ Logistic arrangements have been coordinated	page 6-6
❏ Documents for the initial document review have been requested	page 6-7
❏ The initial interview schedule has been coordinated	page 6-8
❏ Agenda for the Initial Organization Meeting has been coordinated	page 6-9

Entry Criteria for
Phase 5: Findings
(➡page 6-33)

❏ The site visit (Phase 4) has been completed. The following outputs of Phase 4 are used in this phase:	
❏ Completed Validation Worksheets	page 6-20
❏ Completed Interview Worksheets	
(Exploratory Interviews) (Consolidation Interviews)	page 6-18, page 6-30
❏ Preliminary Findings	page 6-24
❏ Document Review Working Notes	
(Initial Document Review) (Document Review) (Final Document Review)	page 6-15, page 6-24, page 6-31

Section H.3   Activities for Each Phase of the SCE Method

The following checklists may be used during each phase of the SCE method to verify that the activities for that phase have been completed.

Activities for
Phase 1: Evaluation
Start (➥page 4-14)

❑ Develop Target Product Profile (Step 1)	page 4-15
❑ Determine Target Process Capability (Step 2)	page 4-16
❑ Select SCE Team (Step 3)	page 4-18

Activities for
Phase 2: General
Preparation
(➥page 5-2)

❑ Create Experience Table (Step 4)
 	
  ❑ Create a Mismatch Identification Table for each development organization
 	
  ❑ Create an Experience Table which summarizes the relevant experience for all development organizations
	page 5-3

❑ Create Critical Subprocess Area List (Step 5)
 	
  ❑ Select the critical subprocess areas to be investigated
 	
  ❑ Prepare a Key Issue Table
	page 5-13

❑ Originate Validation Worksheets (Step 6)	page 5-26

Activities for
Phase 3: Specific
Preparation
(☛page 5-29)

❏ Select Projects to Investigate (Step 7)	page 5-31
❏ Develop Key Issue Worksheet (Step 8)	page 5-34
❏ Record the responses to the Maturity Questionnaire on the Questionnaire Worksheet	
❏ Analyze the Questionnaire Worksheet for inconsistencies and anomalies	
❏ Create a Key Issue Worksheet	
❏ Develop Topic Lists (Step 9)	page 5-39
❏ Individual team members create a list of topics for each critical subprocess area.	
❏ Create a consolidated list of topics for each critical subprocess area.	
❏ Add Topics to Validation Worksheet (Step 10)	page 5-41
❏ Prepare for Exploratory Interviews (Step 11)	page 5-44
❏ Develop the interview plan	
❏ Prepare Interview Worksheets for each interview candidate.	
❏ Prepare Entry Briefing (Step 12)	page 5-52

Activities for Phase 4: Site Data Collection (➡page 6-11)

❑ Conduct Initial Organization Meeting (Step 13)	page 6-13
❑ Conduct Initial Document Review (Step 14)	page 6-15
❑ Conduct Exploratory Interviews (Step 15)	page 6-18
❑ Hold Team Caucus (Step 16)	page 6-20
❑ Conduct Document Review (Step 17)	page 6-24
❑ Develop Preliminary Findings (Step 18)	page 6-24
❑ Create Consolidation Plan (Step 19)	page 6-27
❑ Conduct Consolidation Interviews (Step 20)	page 6-30
❑ Conduct Final Document Review (Step 21)	page 6-31

Activities for Phase 5: Findings (➡page 6-33)

❑ Determine Findings (Step 22)	page 6-34
❑ Produce Findings Report (Step 23)	page 6-36
❑ Conduct Exit Briefing (Step 24)	page 6-38

Section H.4   Coordinating Site Visit Arrangements

The following checklist may be used to verify that the necessary arrangements for a site visit have been coordinated with the organization to be evaluated.

Checklist for
Coordinating Site Visit
Arrangements

At least 2 months prior to site visit	
❑ Negotiate date for site visit	page 6-3
❑ Ask the development organization to identify a point of contact	page 6-5
❑ Notify development organization of logistic requirements	page 6-6
At least 2 weeks prior to the site visit	
❑ Let the development organization know which projects have been selected for evaluation.	
❑ Request documents for initial document review	page 6-7
❑ Coordinate the initial interview plan	page 6-8
❑ Coordinate agenda for initial meeting	page 6-9
❑ Ask the development organization to identify a site visit coordinator	page 6-6

Checklist for
Coordinating Site Visit
Arrangements

At least 2 months prior to site visit	
❏ Negotiate date for site visit	page 6-3
❏ Ask the development organization to identify a point of contact	page 6-5
❏ Notify development organization of logistic requirements	page 6-6
At least 2 weeks prior to the site visit	
❏ Let the development organization know which projects have been selected for evaluation.	
❏ Request documents for initial document review	page 6-7
❏ Coordinate the initial interview plan	page 6-8
❏ Coordinate agenda for initial meeting	page 6-9
❏ Ask the development organization to identify a site visit coordinator	page 6-6

Appendix I   Sample Entry Briefing

This appendix contains an example of an entry briefing that a team might give during the Initial Organization Meeting at the beginning of a site visit. The type of information to be presented in a specific exit briefing should be determined prior to the site visit by the sponsoring agency.

The sample entry briefing included here assumes that the findings from the evaluation will be presented during the exit briefing at the end of the site visit.

Entry Briefing

for

Software Capability Evaluation

Contractor XYZ
Month, Date, 1993

Conducted at the request of <Sponsor Organization & Code>

SCE

Requirement: Contract XXXXXX.XX xx Month, Year

Purpose:

- To evaluate the offeror's Software Process Capability

- SCE Team is not here to discuss other issues
  concerning your offer

Agenda

Introduction of SCE Team Members

Description of the SCE

The On-Site Activities

Schedule of Site Visit Activities

The Ground Rules the Team Intends to Follow

The Processes the Team Will Look At

Sample Findings

Exit Briefing

The SCE Team

Team Leader:
    <name, organization>

Team Members:
    <name, organization>
    <name, organization>
    <name, organization>
    <name, organization>

Team Member Qualifications:

- 12 to 23 years experience in software development and acquisition

- Minimum of 5 years experience in <XYZ application domain>

- Training in the SCE method at the Software Engineering Institute

SCE Description

A method for evaluating the software process of an organization to gain insight into its software development capability.

Five phase evaluation process using the Capability Maturity Model for Software, Version 1.1, as the basis for the evaluation.

Three day site visit conducted by a five person evaluation team.

Outcome: findings on process capability in terms of software development Key Process Areas (KPAs) - strengths, weaknesses, and improvement activities.

No recommendations will be made.

Site Visit Activities

Initial Organization Meeting

Initial Document Review

Exploratory Interviews

Document Review

Team Caucus

Consolidation Interviews

Final Document Review

Prepare Findings

Schedule

Day 1

8:30-10:00	Initial Organization Meeting
10:00-12:00	Initial Document Review
1:00-4:30	Interviews
4:30-?	Team Caucus and Document Review

Day 2

8:30-12:00	Interviews
1:00-2:00	Team Caucus and Document Review
2:00-4:00	Interviews
4:30-?	Team Caucus and Document Review

Day 3

8:30-10:30	Consolidation Interviews
10:30-3:30	Team Caucus and Document Review
3:30-4:00	Exit Briefing

Ground Rules the Team Intends to Follow

Team decisions are made through consensus

The team will interview one individual at a time

There will be no attribution of information obtained to an individual or to a specific project.

The team will look for objective evidence (or lack of objective evidence) to substantiate what it hears in interviews.

The team may interview an individual more than once.

The interview schedule will become dynamic after the first day.

All changes to the interview schedule and requests for initial documentation will be made through the site visit coordinator.

All documentation will be returned at the end of the site visit. The team will not make any copies of the documents.

Processes the Team Will Look At

Processes Are Specified in Terms of the CMM KPAs

Requirements Management

Software Project Planning

Software Project Tracking and Oversight

Software Subcontract Management

Software Quality Assurance

Software Configuration Management

Organization Process Focus

Organization Process Definition

Training Program

Integrated Software Management

Software Product Engineering

Intergroup Coordination

Peer Reviews

Requirements for a Finding

The ⎯⎯ must observe supporting evidence in two or more independent sources.

The team must generate the findings through a consensus process. That is, there are no minority opinions opposed to the findings.

Findings Category Definitions

Strength - A strength indicates a particular part of the software process capability that is sufficiently robust to mitigate the development risks due to software process.

Weakness - A weakness indicates a particular part of the software process capability that has characteristics that increase the risks due to the software process.

Improvement Activity - A process improvement that is not yet institutionalized - for example, a pilot program that implements a new configuration management process. It indicates potential mitigation of risk due to software process.

Key Process Area:
Software Quality Assurance

STRENGTHS:

• Independent Reporting Chain
• Highly Visible
• Insuring Software Engineering Standards Compliance
• Management Commitment - Strong Staff

WEAKNESSES:

• Inconsistent Auditing
• Ineffective Use of Resources

IMPROVEMENT ACTIVITIES:

• Establishing Procedures for Consistent Auditing

Exit Briefing

At the end of the site visit, findings will be presented as a courtesy.

Details will not be discussed at this time.

The final report will be available <date>.

To obtain more information about the results of this evaluation contact:

XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX

Appendix J   Sample Exit Briefing

This appendix contains an example of an exit briefing that a team might give at the end of a site visit. The type of information to be presented in a specific exit briefing should be determined prior to the site visit by the sponsoring agency.

When SCE is used in a source selection application, the Procuring Contract Officer must agree to the agenda of the exit briefing. Many agencies make it a policy not to debrief the findings at the exit briefing. In that case there would be no formal presentation. The team would simply thank the development organization for its support and let them know that the site visit had been concluded.

The sample exit briefing included here does contain the findings from the evaluation. In a real exit briefing there would be a finding sheet for each KPA in the target process capability. In the sample exit briefing, only a few finding sheets are included to show the format.

Exit Briefing

for

Software Capability Evaluation

Contractor XYZ
Month, Date, 1993

Conducted at the request of <Sponsor Organization & Code>

These findings are presented as a courtesy.

Details will not be discussed at this time.

The final report will be available <date>.

To obtain more information about the results
of this evaluation contact:

XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX

Recap of SCE Activities

Interviews

25 Initial, 10 Follow-up

Corporate level through software engineers

Information obtained through interviews was verified through document review

Findings Are Effective If They:

Motivate improvement in contractor process capability,
and benefit the acquisition.

Provide data to help the program office identify, assess,
and reduce program risks.

Enable acquisition organizations to be process oriented without
specifying practices on the contract.

Allow <XYZ organization>, the sponsor, and the contractor to share
in the results and consequences of software processes.

Findings Category Definitions

Strength - A strength indicates a particular part of the software process capability that is sufficiently robust to mitigate the development risks due to software process.

Weakness - A weakness indicates a particular part of the software process capability that has characteristics that increase the risks due to the software process.

Improvement Activity - A process improvement that is not yet institutionalized - for example, a pilot program that implements a new configuration management process. It indicates potential mitigation of risk due to software process.

Requirements for a Finding

The team must observe supporting evidence in two or more independent sources.

The team must generate the findings through a consensus process. That is, there are no minority opinions opposed to the findings.

Key Process Area:
Software Project Tracking and Oversight

STRENGTHS:

• Lead software engineer assigned to all subsystems.

• Periodic status review meetings with key personnel present.

WEAKNESSES:

• Lack of management indicators for project oversight.

• Lack of commitment to using established <organization> processes

IMPROVEMENT ACTIVITIES:

• Implementation of "Product Realization Process".

Key Process Area:
Software Quality Assurance

STRENGTHS:

• SQA organization independent from software development organization.

WEAKNESSES:

• Minimal SQA involvement throughout development and maintenance of subsystem.

• No evidence of verification of <organization> standards and processes.

IMPROVEMENT ACTIVITIES:

• None observed.

Key Process Area:
Software Project Planning

STRENGTHS:

• None observed.

WEAKNESSES:

• Lack of usage of formal estimation process.

• Estimation data not recorded and reused.

IMPROVEMENT ACTIVITIES:

• None observed.

Key Process Area:
Software Configuration Management

STRENGTHS:

• Well-defined and functioning CCB.

WEAKNESSES:

• Lack of early CM involvement in software development.

• No evidence of a mechanism for traceability of developed items.

IMPROVEMENT ACTIVITIES:

• Implementation of <name of CM System>.

Key Process Area:
Peer Reviews

STRENGTHS:

• Independent test and evaluation prior to product release.

• Early involvement of the System Test Activity in the development process.

WEAKNESSES:

• No evidence of implementation of "In-Process Quality Inspection".

• Inadequate process for recording and tracking action items to closure from review meetings.

IMPROVEMENT ACTIVITIES:

• Test organization responsible for writing the test plan / procedures.

Key Process Area:
Training Program

STRENGTHS:

• None observed.

WEAKNESSES:

• No defined job related training requirements for personnel.

IMPROVEMENT ACTIVITIES:

• None observed.

Key Process Area:
Organization Process Definition

STRENGTHS:

• Existence of various <organization> standards, processes, and procedures.

WEAKNESSES:

• Lack of implementation and enforcement of<organization> standards, processes, and procedures.

IMPROVEMENT ACTIVITIES:

• Efforts to tailor and implement "Software Engineering Process" and "Product Realization Process" documents to the software applications.

Key Process Area:
Organization Process Focus

STRENGTHS:

• None observed.

WEAKNESSES:

• Lack of organizational focus to facilitate and implement <organization> software improvement efforts.

IMPROVEMENT ACTIVITIES:

• None observed

Appendix K   Glossary

Acquisition agency: an organization in charge of a government procurement effort. For purposes of this document, an acquisition agency is the sponsoring organization using the SCE method for a source selection.

Applicable standards: a minor attribute used in SCE. This attribute indicates the development standards that are imposed on the project such as DoD-STD-2167A, DoD-STD-2168, or MIL-STD-1521B.

Application of the SCE method: synonym for *use of the SCE method*.

Application domain: a major attribute used in SCE. An application domain is "a bounded set of related systems (i.e., systems that address a particular type of problem). Development and maintenance in an application domain usually requires special skills and/or resources. Examples include payroll and personnel systems, command and control systems, compilers, and expert systems" [Paulk 93b]. For SCE, this is a major attribute used within the various profiles. *The application domain* attribute indicates the area of subject matter expertise needed to translate system requirements into software requirements, and indicates significant differences in the engineering practices which transform the software requirements into accepted code.

Attributes: characteristics of a software product or project. For purposes of an SCE, there are three categories of attributes: major attributes, minor attributes, and schedule attributes. The attributes used in SCE are defined and discussed in Appendix C.

Candidate findings: findings for which there is not yet enough objective evidence to make a decision.

Caucus: SCE teams participate in three types of caucuses, or meetings, during an SCE:

Ongoing team caucus (Step 16): a meeting in which SCE team members analyze, share, and consolidate information in order to reach conclusions about what was seen and heard as a result of probing the implementation of a subprocess area.

Preliminary findings caucus (Step 18): a meeting in which team members articulate conclusions about the subprocess areas based on the information available.

Findings caucus (Step 22): a meeting in which the team analyzes information they have learned to date, including the consolidation interviews and Final Document Review to determine whether the information confirms or negates any of the preliminary findings.

Capability Maturity Model (CMM): "a description of the stages through which software organizations evolve as they define, implement, measure, control, and improve their software processes" [Paulk 93b]. For SCE this is a model consisting of five maturity levels and associated key process areas (KPAs) which are used for evaluating a development organization's software process capability. (See also *maturity model.*)

Common feature: "an attribute that indicates whether the implementation and institutionalization of a key practice is effective, repeatable, and lasting" [Paulk 93b]. There are 5 common features defined for CMM v1.1: commitment to perform, ability to perform, activities performed, measurement and analysis, and verifying implementation.

Configuration management tool: a minor attribute in SCE. This attribute defines the tool set used on the host development system for supporting such activities as the software build process, baselining, and version control.

Contract monitoring: one of the two primary applications of the SCE method. In contract monitoring, SCE can serve as an input for an incentive/award fee or can be used help the sponsoring organization tailor its contract monitoring efforts based on the observed strengths and weaknesses of the development organization's processes.

Critical subprocess area: a subprocess area that is selected by the team for evaluation. A critical subprocess area is selected from within a Target Process Capability KPA. The set of all critical subprocess areas is the Critical Subprocess Area List, and will be investigated at all development organization sites. Collectively, the critical subprocess areas define the scope of the SCE.

Customer: a minor attribute in SCE. This attribute indicates who the development is being done for. Examples include one of the DoD services or a particular market within industry.

Development organization: an organization that develops and/or maintains software products, which is also the recipient of an SCE.

Development organization community: all of the development organizations that are involved with a particular use of the method. In a source selection these are the offerors (or all of the offerors remaining after a competitive range determination), and possibly their subcontractors.

Directive: an order or instruction describing actions that must be performed and authorizing their performance.

Document review: the process of examining documents to find evidence of the processes used by a development organization. Documents can define and standardize processes, can indicate commitment to use the processes, can provide an audit trail of processes that were used, and can collect data about process performance. Three levels of documents are reviewed during an SCE: *organization-level*, *project-level*, and *implementation-level*.

Feature: one of a set of attributes that provide a view of "whether the implementation and institutionalization of a key practice are effective, repeatable, and lasting" [Paulk 93b]. The features used in SCE are a refinement of the common features of CMM v1.1; they include the common features and additional subfeatures derived from the common features. Examples of features are ability to perform, organizational structures, training, plans and procedures, etc. Features are defined in Appendix B. (See common feature.)

Final findings: output from executing the SCE method. Final findings are used to develop the formal findings report.

Findings: includes preliminary findings, candidate findings and final findings. Findings are strengths, weaknesses, or improvement activities. In some cases, an explicit finding of "no finding" can be generated. For example, if there are no subcontractors planned to be used for a development, and no subcontractors are involved with the projects that are evaluated, then a "no finding" would result for the subprocess areas that deal with subcontractor management.

Host development system: a minor attribute in SCE. This attribute refers to the computer environment which will be used for the software development.

Implementation-level documents: the third of three levels of documents reviewed during an SCE. These are documents which provide an audit trail of processes that were used, and can be used by the development organization to collect data about process performance.

Improvement activity: a process improvement that is not yet institutionalized—for example, a pilot program that implements a new configuration management process. In SCE, it indicates potential mitigation of risk due to software process.

Interviewing: the process of questioning personnel from the development organization to find evidence of the processes used by the development organization. During an SCE, the SCE team typically interviews one person at a time. Interviews provide insight into how processes are implemented and show the extent to which processes have been internalized by members of the development organization.

Key issue: the relationship between a critical subprocess area on the Critical Subprocess Area List and a development organization or organizations. The subprocess area is a key issue for the development organization

- If there is information known about the development organization that relates it specifically to that critical subprocess area. As examples, this can happen because of a mismatch in the Mismatch Identification Table or because the organizational charts indicate a possible risk. These observations could cause the team to identify a particular subprocess area as a key issue that needs to be probed.

- If the subprocess area has been selected as a key issue for all development organizations. As examples, this could happen because the operational precedence attribute in the Target Product Profile caused the team to identify a subprocess area as a key issue that needed to be probed, or because the subprocess area was part of the nucleus capability.

Key process area (KPA): "a cluster of related activities that, when performed collectively, achieve a set of goals considered important for establishing process capability" [Paulk 93b]. Each KPA contributes to the environment in which development organizations create software products. Within the CMM, the KPAs are organized into five basic levels of process maturity to describe the progression from an ad hoc software process to one that is well defined and can act as a stable foundation for continuous process improvement.

Language(s): a minor attribute for SCE. This attribute indicates the programming languages in which the code is to be written, or in which it has been written.

Mapping: the relationship between actual practices in the software process implementation and the KPAs.

Maturity level: "a well-defined evolutionary plateau toward achieving a mature software process" [Paulk 93b].

Maturity model: for SCE, this is a model consisting of five maturity levels and associated Key Process Areas (KPAs) which are used for evaluating a development organization's software process capability. The maturity model used in Version 2.0 of the SCE method is defined in the *Capability Maturity Model for Software, Version 1.1* [Paulk 93a].

Operational Precedence: a major attribute used in SCE. This attribute indicates whether the end user has previous experience with the type of system to be built. Systems that are providing a new capability tend to have more changes to the requirements than do ones that are replacing existing systems.

Organization-level documents: the first (or top) level of three levels of documents reviewed during an SCE. These are the policies and procedures which establish the development environment for all company project

activities. Organizational level documents define the process and management constraints the organization places on projects.

Policy: "a guiding principle, typically established by senior management, adopted by an organization to influence and determine decisions" [Paulk 93b].

Preliminary findings: findings for a subprocess area generated during caucus. These represent SCE team consensus about a subprocess area or *KPA*, and remove the area from further consideration during the site visit. These are the basis for the final findings.

Procedure: a written description of a course of action to be taken to perform a given task [IEEE 91].

Process capability: "the range of expected results that can be achieved by following a process" [Paulk 93b].

Product Type: a major attribute in SCE. The product type attribute refers to the particular aspect of the application domain which the system will support or to the type of service which the system will provide. For example, displays or communications could be product types in a command and control system, a weapons system, or another application domain. Although there may be similarities in the communications subsystem in the various application domains, they each have their own set of unique problems which must be addressed.

Profiles: a profile is the set of attributes (such as the major attributes Application Domain, Product Type, and Size) associated with a software product and the project that develops the product. There are three types of profiles used in SCE: Target Product Profiles, Proposed Project Profiles, and Project Profiles. The Target Product Profile represents the "customer view" of the product to be built, and captures the attributes of the desired product. The Proposed Project

Profile represents the development organization's view of the planned development. Project Profiles represent the actual attributes of ongoing or recently completed projects.

Project-level documents: the second of three levels of documents reviewed during an SCE. These are documents which define the development processes in use for a particular project. Project level documents define the detailed processes that are used to manage, coordinate, and integrate the engineering activities required for the development.

Project Profile: see *Profiles.*

Proposed Project Profile: see *Profiles.*

Request for Proposal (RFP): an acquisition document that describes characteristics of the system the government wants to acquire. This document is used to solicit proposals from commercial development organizations (offerors) and to communicate the characteristics of the desired system to the offerors. In source selection, this is the document that specifies that an SCE will be performed.

Results: how the findings are used by the sponsoring organization—for example, in risk determination for source selection.

SCE Method: a method for evaluating the software process of an organization to gain insight into its software development capability.

Scope of SCE: the boundaries of the investigation, in terms of critical subprocess areas within the KPAs in the Target Process Capability. Items outside the defined scope of the SCE can't be looked at during source selection.

Site visit: an investigation conducted by four to six government personnel (the SCE team) over a three day period at a development organization's facility.

Size: a major attribute for SCE. The size attribute indicates the magnitude of the product (and hence the required project). Size is composed of three related attributes. The *contract duration* is the estimated or required length of time for the development of the software product. The *software team size* is the number of software developers who will be involved in the project. The *estimated software size* is the amount of code to be developed.

Software development plan (SDP): "the collection of plans that describe the activities to be performed for the software project" [Paulk 93b]. This could be, but is not necessarily the same document referred to in DoD-STD-2167A.

Software process implementation: a tailored set of practices that defines how software development work is supposed to be done.

Software process capability: "the range of expected results that can be achieved by following a process" [Paulk 93b]. For purposes of an SCE, those CMM-related processes which provide a detailed environment for one or more development teams to produce software products. The processes evaluated include decision making processes (such as software project planning) and communication processes (such as peer reviews).

Source selection: one of the two primary applications of the SCE method. In source selection, the results of the SCE are used by the sponsoring organization to characterize the software process-related risk of awarding a contract to an offeror. SCE is only one criterion among many used to select software contractors in government acquisitions.

Sponsoring organization: the organization that commissions the SCE to be performed and uses the findings.

Standard: "mandatory requirements employed and enforced to prescribe a disciplined, uniform approach to software development" [Paulk 93b].

Strength: in SCE, strength indicates a particular part of the software process capability that is sufficiently robust to mitigate the development risks due to software process.

Subcontractor: a development organization that is contracted to work for another development organization to produce software products.

Subcontractors: a major attribute in SCE. This attribute is used to indicate whether the development organization intends to use subcontractors in the development, and is a factor if they lack experience with subcontract management.

Subprocess area: a set of activities in an implemented process that, acting together, attempts to achieve one of the goals of a KPA. Alternatively, a focused subset of process activities that work toward achieving a specific KPA goal. This is a subdivision of a KPA that addresses a major process activity within the larger cluster of related activities that make up the KPA. The KPA goals represent desired states; subprocess areas encapsulate the activities needed to achieve those states. The Critical Subprocess Area List is a set of subprocess areas which collectively define the scope of the SCE.

Target: a minor attribute in SCE. This attribute indicates the hardware configuration that the developed software will run on when operational.

Target Process Capability: the process capability that is most appropriate for the planned development; the process capability desired by the sponsoring organization for the product to be developed. The Target Process capability consists of a set of KPAs, and establishes the boundaries of the SCE investigation—a KPA is evaluated if and only if it is part of the Target Process Capability.

Target Product Profile: see *Profiles.*

Topic: a topic defines a subject that will be probed during the investigation. A topic is an abstraction of a work practice that corresponds to a portion of the process implementation for the development organization. Topics are intended to be detailed enough to focus the investigation on observable, documented work practices, but sufficiently abstract that they avoid prescribing how the subprocess area is implemented. Topics are selected by considering the features associated with each subprocess area.

Type of Work: a major attribute for SCE. This attribute indicates the portion of the development life cycle which will be performed. As examples of different types of work, in "full software development" a development organization is required to build a product based on the system requirements, while in "code development only" the development organization is required to develop code according to the system requirements and software top level design provided by the issuing authority.

Use of the SCE method: executing the SCE method within a particular context. To date, the two primary uses of the SCE method are in source selection and contract monitoring. This is sometimes referred to as the application of the method.

Weakness: In SCE, weakness indicates a particular part of the software process capability that has characteristics that increase the risks due to software process.

Appendix L   Bibliography

[Humphrey 89]   Humphrey, W. S. *Managing the Software Process.* Reading, Ma.: Addison-Wesley, 1989.

[IEEE 91]   ANSI/IEEE Std 610.12-1990, *IEEE Standard Glossary of Software Engineering Terminology.* February 1991.

[Paulk 93a]   Paulk, Mark C., et al. *Capability Maturity Model for Software, Version 1.1 (CMU/SEI-93-TR-24).* Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1993.

[Paulk 93b]   Paulk, Mark C., et al. *Key Practices of the Capability Maturity Model, Version 1.1 (CMU/SEI-93-TR-25).* Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1993.

[SCE 93a]   Members of the SCE Project, *Software Capability Evaluation Version 2.0 Implementation Guide* (CMU/SEI-94-TR-05). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1994.

[SCE 93b]   Members of the SCE Project, *Software Capability Evaluation Versirn 2.0 Method Description* (CMU/SEI-94-TR-06). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, 1994.

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS None		
2a. SECURITY CLASSIFICATION AUTHORITY N/A		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release Distribution Unlimited		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A				
4. PERFORMING ORGANIZATION REPORT NUMBER(S) CMU/SEI-94-HB02		5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION Software Engineering Institute	6b. OFFICE SYMBOL (if applicable) SEI	7a. NAME OF MONITORING ORGANIZATION SEI Joint Program Office		
6c. ADDRESS (city, state, and zip code) Carnegie Mellon University Pittsburgh PA 15213		7b. ADDRESS (city, state, and zip code) HQ ESC/ENS 5 Eglin Street Hanscom AFB, MA 01731-2116		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION SEI Joint Program Office	8b. OFFICE SYMBOL (if applicable) ESC/ENS	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F1962890C0003		
8c. ADDRESS (city, state, and zip code)) Carnegie Mellon University Pittsburgh PA 15213		10. SOURCE OF FUNDING NOS.		

		PROGRAM ELEMENT NO 63756E	PROJECT NO. N/A	TASK NO N/A	WORK UNIT NO. N/A

11. TITLE (Include Security Classification)
Software Capability Evaluation (SCE) Version 2.0 Team Members' Guide

12. PERSONAL AUTHOR(S)
Members of the CMM-Based Appraisal Project

13a. TYPE OF REPORT Final	13b. TIME COVERED FROM TO	14. DATE OF REPORT (year, month, day) October 1994	15. PAGE COUNT 255 pp.

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB. GR.

18. SUBJECT TERMS (continue on reverse if necessary and identify by block number)
software capability evaluation (SCE)
capability maturity model (CMM)
software process

19. ABSTRACT (continue on reverse if necessary and identify by block number)

Software Capability Evaluation (SCE) is a method for independently evaluating the software process of an organization to gain insight into its software development capability. The method is defined in the report "Software Capability Evaluation Version 2.0 Method Description." [SCE 93b]

This document, the "Software Capability Evaluation Version 2.0 Team Members' Guide," is intended for use by members of teams that will be conducting an SCE. The guide provides detailed step-by-step instructions and heuristic information to assist an SCE team in preparing for and conducting an evaluation.

(please turn over)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ■ SAME AS RPT □ DTIC USERS ■	21. ABSTRACT SECURITY CLASSIFICATION Unclassified, Unlimited Distribution	
22a. NAME OF RESPONSIBLE INDIVIDUAL Thomas R. Miller, Lt Col, USAF	22b. TELEPHONE NUMBER (include area code) (412) 268-7631	22c. OFFICE SYMBOL ESC/ENS (SEI)

ABSTRACT — continued from page one, block 19

ABSTRACT — continued from page one, block 19